



**ESCUELA POLITÉCNICA SUPERIOR**

**INGENIERÍA TÉCNICA**

**EN INFORMÁTICA DE GESTIÓN**

**PROYECTO FIN DE CARRERA**

**Aplicación para la automatización de simulaciones  
en diferentes tipologías de redes neuronales para  
la resolución de problemas multidominio**

**Application for the automation of simulations in  
different types of neural networks to solving  
multidomain problems**

Autor: Miguel Fortes Gómez

Tutor: Israel González Carrasco

Noviembre 2010



## **Agradecimientos**

En primer lugar, quiero agradecer a mis padres todo lo que me han dado. El apoyo que he recibido durante la realización de este proyecto. Sin ellos no podría haberlo hecho. También agradecer el apoyo de mi hermano.

Durante mi etapa universitaria he conocido muchos amigos y compañeros. Hemos pasado muchas horas juntos en las que ha habido muy buenos momentos. Quedarán en mi memoria gratos recuerdos.

Agradecer también a mi tutor por su dedicación en este proyecto.

Agradecer a mi novia por animarme en todo momento a continuar y no desistir. Gracias por ser como eres, no cambies nunca.

---



## Summary

One of the most common problems in Neurocomputation is the training neural networks, especially when it comes to training a large network which turns out to be very costly in terms of time, cpu resources, etc.

The application that was developed in this project facilitates the training of different neural networks architectures. For this it has automated the process of learning, creating, developing and evaluating a series of experiments that can be run sequentially and iteratively in multiple neural networks. The designed system is Multi-domain, which can work with data in any field of research.

The designed system provides the treatment of the data. The application allows to extract data from an Excel file and to treat data under various division techniques to generate experiments.

The developed application allows to identify different neural network architectures. Automatically, the system performs the identification to select the folder that contains files containing neural networks.

In order to facilitate viewing of the training result of the experiments on the different architectures of neural networks, the application can generate a file of assessment for each experiment and a summary file for the entire simulation. These files allow you to extract relevant information from the simulation for a clearer display of results.

For evaluation and validation of the created system it has been performed several tests on different neural network architectures applied in various domains.

---



## **Resumen**

Uno de los problemas más comunes en neurocomputación es el entrenamiento de las redes neuronales, sobretodo cuando se trata de entrenar una red de grandes dimensiones, ya que esto puede resultar muy costoso en cuanto a tiempo, recursos de cpu etc.

La aplicación que se ha desarrollado en este proyecto facilita el entrenamiento de diferentes arquitecturas de redes neuronales. Para ello se ha automatizado el proceso de aprendizaje, creando, desarrollando y evaluando series de experimentos que pueden ejecutarse de manera secuencial e iterativa sobre múltiples redes neuronales. El sistema diseñado es multidominio, lo que permite trabajar con datos de cualquier ámbito de investigación.

El sistema diseñado contempla el tratamiento de los datos. La aplicación permite extraer los datos de un fichero en Excel y tratarlos bajo varias técnicas de división para generar los experimentos.

La aplicación desarrollada permite identificar diferentes arquitecturas de redes neuronales. Automáticamente, el sistema realiza la identificación al indicar la carpeta que contiene los ficheros que contiene las redes neuronales.

Con el objetivo de facilitar la visualización del resultado del entrenamiento de los experimentos en las diferentes arquitecturas de redes neuronales, la aplicación permite generar un fichero de evaluación por cada experimento y un fichero de resumen para el conjunto de la simulación. Estos ficheros permiten extraer la información relevante de la simulación para una visualización mas clara de los resultados.

Para la evaluación y validación del sistema creado se han realizado pruebas sobre diferentes arquitecturas de red neuronal aplicadas en varios dominios.

---





## **Tabla de contenidos**

<b>1</b>	<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>2</b>	<b>DEFINICIONES Y ACRÓNIMOS.....</b>	<b>4</b>
2.1	DEFINICIONES.....	4
2.2	ACRÓNIMOS .....	8
<b>3</b>	<b>ESTUDIO DE VIABILIDAD DEL SISTEMA.....</b>	<b>9</b>
3.1	INTRODUCCIÓN.....	9
3.1.1	<i>Propósito del plan .....</i>	<i>9</i>
3.2	ESTABLECIMIENTO DEL ALCANCE DEL SISTEMA .....	10
3.2.1	<i>Estudio de la solicitud .....</i>	<i>10</i>
3.2.2	<i>Identificación del alcance del sistema.....</i>	<i>11</i>
3.2.3	<i>Identificación de los interesados en el sistema (stakeholders). .....</i>	<i>12</i>
3.3	ESTUDIO DE LA SITUACIÓN ACTUAL .....	13
3.3.1	<i>Valoración del estudio de la situación actual.....</i>	<i>13</i>
3.3.2	<i>Realización del diagnóstico actual.....</i>	<i>13</i>
3.4	DEFINICIÓN DE LOS REQUISITOS DEL SISTEMA.....	14
3.4.1	<i>Especificación de los casos de uso .....</i>	<i>15</i>
3.4.2	<i>Definición de los requisitos del sistema.....</i>	<i>25</i>
3.5	ESTUDIO DE ALTERNATIVAS DE SOLUCIÓN .....	39
3.6	VALORACIÓN DE LAS ALTERNATIVAS .....	40
3.7	SELECCIÓN DE LA SOLUCIÓN.....	40
<b>4</b>	<b>GESTIÓN DEL PROYECTO.....</b>	<b>41</b>
4.1	CICLO DE VIDA .....	41

---



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

4.2	ORGANIZACIÓN DEL PROYECTO.....	44
4.3	PLANIFICACIÓN.....	50
4.4	ESTIMACIÓN DE COSTES .....	57
<b>5</b>	<b>ANÁLISIS DEL SISTEMA .....</b>	<b>59</b>
5.1	INTRODUCCIÓN.....	59
5.1.1	<i>Objetivo del Análisis del Sistema .....</i>	<i>59</i>
5.1.2	<i>Alcance.....</i>	<i>59</i>
5.2	DEFINICIÓN DEL SISTEMA .....	60
5.2.1	<i>Determinación del alcance del sistema .....</i>	<i>60</i>
5.2.2	<i>Identificación del entorno tecnológico.....</i>	<i>60</i>
5.2.3	<i>Especificación de estándares y normas.....</i>	<i>61</i>
5.3	ESTABLECIMIENTO DE REQUISITOS SOFTWARE.....	64
5.3.1	<i>Obtención de requisitos.....</i>	<i>64</i>
5.4	IDENTIFICACIÓN DE SUBSISTEMAS DE ANÁLISIS.....	82
5.5	ANÁLISIS DE LOS CASOS DE USO.....	83
5.5.1	<i>Identificación de clases asociadas a un caso de uso.....</i>	<i>84</i>
5.5.2	<i>Descripción de la interacción de objetos .....</i>	<i>85</i>
5.6	ANÁLISIS DE CLASES .....	86
5.6.1	<i>Identificación de responsabilidades y atributos .....</i>	<i>87</i>
5.6.2	<i>Identificación de asociaciones .....</i>	<i>92</i>
5.6.3	<i>Identificación de generalizaciones.....</i>	<i>95</i>
5.7	DEFINICIÓN DE LAS INTERFACES DE USUARIO .....	96
5.7.1	<i>Especificación de principios generales de la interfaz .....</i>	<i>96</i>

---



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

5.7.2	<i>Identificación de perfiles y diálogos.....</i>	96
5.7.3	<i>Especificación de formatos individuales de la interfaz de la pantalla</i>	96
5.8	ANÁLISIS DE CONSISTENCIA Y ESPECIFICACIÓN DE REQUISITOS .....	103
5.8.1	<i>Verificación y análisis de consistencia entre modelos .....</i>	103
5.8.2	<i>Validación de los modelos.....</i>	105
<b>6</b>	<b>DISEÑO DEL SISTEMA .....</b>	<b>106</b>
6.1	INTRODUCCIÓN.....	106
6.1.1	<i>Propósito .....</i>	106
6.2	ALCANCE .....	107
6.3	DEFINICIÓN DE LA ARQUITECTURA .....	109
6.3.1	<i>Definición de los niveles de la arquitectura.....</i>	109
6.3.2	<i>Especificación de estándares y normas de diseño y construcción</i>	109
6.3.3	<i>Identificación de los subsistemas de la aplicación.....</i>	111
6.3.4	<i>Especificación del entorno tecnológico.....</i>	111
6.3.5	<i>Especificación de requisitos de seguridad y operación.....</i>	112
6.3.6	<i>Estudio de la seguridad requerida en el proceso de diseño del sistema</i>	112
6.3.7	<i>Análisis de los riesgos del entorno tecnológico.....</i>	112
6.4	DISEÑO DE LA ARQUITECTURA DE SOPORTE.....	113
6.5	DISEÑO DE CASOS DE USO REALES .....	115
6.5.1	<i>Identificación de clases asociadas a un caso de uso.....</i>	115
6.5.2	<i>Diseño de la realización de los casos de uso.....</i>	116

---



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

6.6	DISEÑO DE CLASES .....	118
6.6.1	<i>Identificación de clases de diseño .....</i>	<i>118</i>
6.6.2	<i>Identificación de atributos y métodos de clases de diseño .....</i>	<i>120</i>
6.7	DISEÑO FÍSICO DE DATOS .....	130
6.8	VERIFICACIÓN Y ACEPTACIÓN DE LA ARQUITECTURA DEL SISTEMA .....	131
6.8.1	<i>Análisis de consistencia de las especificaciones de diseño.....</i>	<i>131</i>
6.9	ESPECIFICACIÓN TÉCNICA DEL PLAN DE PRUEBAS .....	132
6.10	ESTABLECIMIENTO DE REQUISITOS DE IMPLANTACIÓN .....	133
6.10.1	<i>Especificación de Requisitos de Implantación .....</i>	<i>133</i>
<b>7</b>	<b>PLAN DE VALIDACIÓN Y VERIFICACIÓN DEL SOFTWARE .....</b>	<b>135</b>
7.1	INTRODUCCIÓN.....	135
7.1.1	<i>Propósito del documento .....</i>	<i>135</i>
7.2	PLAN DE PRUEBA .....	136
7.2.1	<i>Definición del Alcance de las pruebas .....</i>	<i>136</i>
7.2.2	<i>Entregables.....</i>	<i>137</i>
7.2.3	<i>Tareas de prueba .....</i>	<i>138</i>
7.2.4	<i>Necesidades del entorno .....</i>	<i>139</i>
7.2.5	<i>Criterio de aceptación/rechazo de un caso de prueba .....</i>	<i>139</i>
7.3	DEFINICIÓN DE PRUEBAS.....	140
7.3.1	<i>Definición de pruebas de Aceptación del sistema.....</i>	<i>140</i>
7.3.2	<i>Definición de las Pruebas Unitarias .....</i>	<i>167</i>
7.3.3	<i>Definición de las pruebas de implantación.....</i>	<i>167</i>
7.3.4	<i>Definición de las pruebas de Integración.....</i>	<i>167</i>

---



7.3.5	<i>Definición de las pruebas de Sistema</i> .....	168
7.4	PLANTILLA DE INFORME DE PRUEBAS.....	173
<b>8</b>	<b>CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN</b> .....	<b>174</b>
8.1	GENERACIÓN DEL CÓDIGO .....	174
8.2	EJECUCIÓN DE LAS PRUEBAS UNITARIAS .....	176
8.3	EJECUCIÓN DE LAS PRUEBAS DE INTEGRACIÓN .....	176
8.4	EJECUCIÓN DE LAS PRUEBAS DE SISTEMA .....	176
8.4.1	<i>Dominio A</i> .....	176
8.4.2	<i>Dominio B</i> .....	179
<b>9</b>	<b>CONCLUSIONES</b> .....	<b>181</b>
<b>10</b>	<b>BIBLIOGRAFÍA</b> .....	<b>182</b>
<b>11</b>	<b>ANEXO A: MANUAL DE USUARIO APLICACIÓN TRATAMIENTO Y ENTRENAMIENTO EN REDES NEURONALES</b> .....	<b>184</b>
11.1	ARRANQUE DE LA APLICACIÓN.....	184
11.2	LOCALIZACIÓN DE FICHEROS .....	186
11.2.1	<i>Fichero de entrada</i> .....	186
11.2.2	<i>Especificar una ruta de salida</i> .....	187
11.3	IMPORTAR DATOS .....	189
11.4	CONFIGURACIÓN DE EXPERIMENTOS.....	193
11.4.1	<i>Técnica de división</i> .....	193
11.4.2	<i>Número de pruebas</i> .....	195
11.4.3	<i>Conjuntos de instancias</i> .....	197
11.4.4	<i>Reparto de instancias</i> .....	198
11.4.5	<i>Añadir experimentos a preparar</i> .....	200

---



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

11.4.6	<i>Añadir experimentos de forma automatizada</i> .....	201
11.4.7	<i>Eliminar experimentos</i> .....	203
11.5	GENERACIÓN DE FICHEROS .....	205
11.6	GESTIÓN DE LA CONFIGURACIÓN .....	208
11.6.1	<i>Guardar configuración</i> .....	208
11.6.2	<i>Cargar configuración</i> .....	210
11.7	INTERFAZ ENTRENAMIENTO .....	212
11.8	LOCALIZACIÓN FICHEROS RED.....	214
11.9	CAMBIAR MANUALMENTE LA TIPOLOGÍA DE RED .....	216
11.10	CONFIGURAR PARÁMETROS DE ENTRENAMIENTO .....	218
11.10.1	<i>Criterio de parada</i> .....	218
11.10.2	<i>Generación de informes</i> .....	220
11.11	ENTRENAMIENTO.....	222
11.12	INFORMES DE EVALUACIÓN Y RESUMEN .....	225
11.12.1	<i>Informes de Evaluación</i> .....	225
11.12.2	<i>Informe de Resumen</i> .....	228
11.13	SALIR DE LA APLICACIÓN .....	230
12	<b>ANEXO B: FORMATO PARA FICHEROS DE ENTRADA .....</b>	<b>231</b>
13	<b>ANEXO C. MANUAL DE USUARIO PARA LA CREACIÓN DE UNA RED NEURONAL EN NEUROSOLUTIONS.....</b>	<b>232</b>

---

---



---

---

## Índice de tablas

TABLA 1. CASO DE USO CU-001 .....	17
TABLA 2. CASO DE USO CU-002 .....	17
TABLA 3. CASO DE USO CU-003 .....	18
TABLA 4. CASO DE USO CU-004 .....	18
TABLA 5. CASO DE USO CU-005 .....	19
TABLA 6. CASO DE USO CU-006 .....	19
TABLA 7. CASO DE USO CU-007 .....	20
TABLA 8. CASO DE USO CU-008 .....	20
TABLA 9. CASO DE USO CU-009 .....	21
TABLA 10. CASO DE USO CU-010 .....	21
TABLA 11. CASO DE USO CU-011 .....	22
TABLA 12. CASO DE USO CU-012 .....	22
TABLA 13. CASO DE USO CU-013 .....	23
TABLA 14. CASO DE USO CU-014 .....	23
TABLA 15. CASO DE USO CU-015 .....	24
TABLA 16. CASO DE USO CU-016 .....	24
TABLA 17. REQUISITO DE USUARIO RU-C-001 .....	27
TABLA 18. REQUISITO DE USUARIO RU-C-002 .....	27
TABLA 19. REQUISITO DE USUARIO RU-C-003 .....	28
TABLA 20. REQUISITO DE USUARIO RU-C-004 .....	28
TABLA 21. REQUISITO DE USUARIO RU-C-005 .....	29
TABLA 22. REQUISITO DE USUARIO RU-C-006 .....	29
TABLA 23. REQUISITO DE USUARIO RU-C-007 .....	30
TABLA 24. REQUISITO DE USUARIO RU-C-008 .....	30
TABLA 25. REQUISITO DE USUARIO RU-C-009 .....	31
TABLA 26. REQUISITO DE USUARIO RU-C-010 .....	31
TABLA 27. REQUISITO DE USUARIO RU-C-011 .....	32
TABLA 28. REQUISITO DE USUARIO RU-C-012 .....	32
TABLA 29. REQUISITO DE USUARIO RU-C-013 .....	33
TABLA 30. REQUISITO DE USUARIO RU-C-014 .....	33
TABLA 31. REQUISITO DE USUARIO RU-C-015 .....	34
TABLA 32. REQUISITO DE USUARIO RU-C-016 .....	34
TABLA 33. REQUISITO DE USUARIO RU-C-017 .....	35
TABLA 34. REQUISITO DE USUARIO RU-C-018 .....	35
TABLA 35. REQUISITO DE USUARIO RU-R-001 .....	36
TABLA 36. REQUISITO DE USUARIO RU-R-002 .....	36
TABLA 37. REQUISITO DE USUARIO RU-R-003 .....	37
TABLA 38. REQUISITO DE USUARIO RU-R-004 .....	37

---

---



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

TABLA 39. REQUISITO DE USUARIO RU-R-005 .....	38
TABLA 40. REQUISITO DE USUARIO RU-R-06 .....	38
TABLA 41. COSTE LICENCIAS.....	57
TABLA 42. COSTE RECURSOS HUMANOS.....	57
TABLA 43. COSTE TOTAL DEL PROYECTO.....	58
TABLA 44. REQUISITO SOFTWARE RS-F-001.....	66
TABLA 45. REQUISITO SOFTWARE RS-F-002.....	66
TABLA 46. REQUISITO SOFTWARE RS-F-003.....	67
TABLA 47. REQUISITO SOFTWARE RS-F-004.....	67
TABLA 48. REQUISITO SOFTWARE RS-F-005.....	68
TABLA 49. REQUISITO SOFTWARE RS-F-006.....	68
TABLA 50. REQUISITO SOFTWARE RS-F-007.....	69
TABLA 51. REQUISITO SOFTWARE RS-F-008.....	69
TABLA 52. REQUISITO SOFTWARE RS-F-009.....	70
TABLA 53. REQUISITO SOFTWARE RS-F-010.....	70
TABLA 54. REQUISITO SOFTWARE RS-F-011.....	71
TABLA 55. REQUISITO SOFTWARE RS-F-012.....	71
TABLA 56. REQUISITO SOFTWARE RS-F-013.....	72
TABLA 57. REQUISITO SOFTWARE RS-F-014.....	72
TABLA 58. REQUISITO SOFTWARE RS-F-015.....	73
TABLA 59. REQUISITO SOFTWARE RS-F-016.....	73
TABLA 60. REQUISITO SOFTWARE RS-F-017.....	74
TABLA 61. REQUISITO SOFTWARE RS-F-018.....	74
TABLA 62. REQUISITO SOFTWARE RS-F-019.....	75
TABLA 63. REQUISITO SOFTWARE RU-F-020 .....	75
TABLA 64. REQUISITO SOFTWARE RU-F-021 .....	76
TABLA 65. REQUISITO SOFTWARE RU-F-022 .....	76
TABLA 66. REQUISITO SOFTWARE RS-I-001 .....	77
TABLA 67. REQUISITO SOFTWARE RS-O-001 .....	77
TABLA 68. REQUISITO SOFTWARE RS-O-002.....	78
TABLA 69. REQUISITO SOFTWARE RS-O-003 .....	78
TABLA 70. REQUISITO SOFTWARE RS-O-004.....	79
TABLA 71. REQUISITO SOFTWARE RS-O-005.....	79
TABLA 72. REQUISITO SOFTWARE RS-D-001 .....	80
TABLA 73. REQUISITO SOFTWARE RS-C-001 .....	80
TABLA 74. REQUISITO SOFTWARE RS-C-002 .....	81
TABLA 75. CLASES ASOCIADAS A CASOS DE USO .....	84
TABLA 76. CLASE INTERFAZ TRATAMIENTO.....	88
TABLA 77. CLASE INTERFAZ IMPORTACIÓN.....	88
TABLA 78. CLASE INTERFAZ ESPERA.....	88

---





## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

TABLA 79. CLASE GESTORCONFIGURACION .....	88
TABLA 80. CLASE GESTOREXPERIMENTO .....	88
TABLA 81. CLASE DIVISIONKFCV .....	89
TABLA 82. CLASE DIVISIONLOOCV .....	89
TABLA 83. CLASE DIVISIONRBP.....	89
TABLA 84. CLASE DIVISIONRBPFIJO .....	90
TABLA 85. CLASE DIVISIONVKFCV .....	90
TABLA 86. CLASE INTERFAZENTRENAMIENTO.....	91
TABLA 87. CLASE GESTORFICHERORED .....	91
TABLA 88. CLASE INTERFAZINFOENTRENAMIENTO .....	91
TABLA 89. CLASE GESTOREVALUACIONRESUMEN .....	91
TABLA 90. ASOCIACIÓN CLASES INTERFAZTRATAMIENTO E INTERFAZIMPORTACIÓN. ....	92
TABLA 91. ASOCIACIÓN CLASES INTERFAZTRATAMIENTO E INTERFAZESPERA. ....	92
TABLA 92. ASOCIACIÓN CLASES INTERFAZTRATAMIENTO Y GESTORCONFIGURACIÓN. ....	92
TABLA 93. ASOCIACIÓN CLASES INTERFAZTRATAMIENTO Y GESTOREXPERIMENTO. ....	93
TABLA 94. ASOCIACIÓN CLASES GESTOREXPERIMENTO E DIVISIONLOOCV.....	93
TABLA 95. ASOCIACIÓN CLASES GESTOREXPERIMENTO E DIVISIONRBP .....	93
TABLA 96. ASOCIACIÓN CLASES GESTOREXPERIMENTO E DIVISIONKFCVFIJO.....	93
TABLA 97. ASOCIACIÓN CLASES GESTOREXPERIMENTO E DIVISIONKFCV .....	93
TABLA 98. ASOCIACIÓN CLASES INTERFAZTRATAMIENTO E INTERFAZENTRENAMIENTO.....	94
TABLA 99. ASOCIACIÓN CLASES INTERFAZENTRENAMIENTO Y GESTORFICHERORED.....	94
TABLA 100. ASOCIACIÓN CLASES INTERFAZENTRENAMIENTO E INTERFAZINFOENTRENAMIENTO .....	94
TABLA 101. ASOCIACIÓN CLASES INTERFAZENTRENAMIENTO E INTERFAZIMPORTACIÓN. ....	94
TABLA 102. MATRIZ DE TRAZABILIDAD REQUISITOS DE USUARIO - CASOS DE USO .....	104
TABLA 103. MATRIZ DE TRAZABILIDAD REQUISITOS SOFTWARE - CASOS DE USO .....	105
TABLA 104. COMPONENTE C-001: REDES NEURONALES .....	114
TABLA 105. CLASES DE DISEÑO ASOCIADAS A CASOS DE USO.....	116
TABLA 106.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE INTERFAZTRATAMIENTO.....	120
TABLA 107.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE INTERFAZIMPORTACIÓN .....	121
TABLA 108.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE INTERFAZESPERA .....	121
TABLA 109.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE GESTORCONFIGURACION .....	122
TABLA 110.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE GESTOREXPERIMENTO .....	122
TABLA 111.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE DIVISIONKFCV .....	123
TABLA 112.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE DIVISIONLOOCV .....	124
TABLA 113.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE DIVISIONRBP.....	125
TABLA 114.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE DIVISIONRBPFIJO .....	126
TABLA 115.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE DIVISIONVKFCV.....	127
TABLA 116.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE INTERFAZENTRENAMIENTO .....	128
TABLA 117.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE GESTORFICHERORED .....	129
TABLA 118.ATRIBUTOS, FUNCIONES Y PROCEDIMIENTOS CLASE GESTOREVALUACIONRESUMEN .....	129

---



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

TABLA 119. MATRIZ DE TRAZABILIDAD CLASES DE DISEÑO - CASOS DE USO.....	131
TABLA 120. REQUISITO DE SOFTWARE RS-IMP-001.....	133
TABLA 121. REQUISITO DE SOFTWARE RS-IMP-002.....	134
TABLA 122. REQUISITO DE SOFTWARE RS-IMP-003.....	134
TABLA 123. PRUEBA PR-A-001 .....	140
TABLA 124. PRUEBA PR-A-002 .....	141
TABLA 125. PRUEBA PR-A-003 .....	141
TABLA 126. PRUEBA PR-A-004 .....	141
TABLA 127. PRUEBA PR-A-005 .....	142
TABLA 128. PRUEBA PR-A-006 .....	142
TABLA 129. PRUEBA PR-A-007 .....	142
TABLA 130. PRUEBA PR-A-008 .....	143
TABLA 131. PRUEBA PR-A-009 .....	143
TABLA 132. PRUEBA PR-A-010 .....	143
TABLA 133. PRUEBA PR-A-011 .....	144
TABLA 134. PRUEBA PR-A-012 .....	144
TABLA 135. PRUEBA PR-A-013 .....	144
TABLA 136. PRUEBA PR-A-014 .....	145
TABLA 137. PRUEBA PR-A-015 .....	145
TABLA 138. PRUEBA PR-A-016 .....	146
TABLA 139. PRUEBA PR-A-017 .....	146
TABLA 140. PRUEBA PR-A-018 .....	146
TABLA 141. PRUEBA PR-A-019 .....	147
TABLA 142. PRUEBA PR-A-020 .....	147
TABLA 143. PRUEBA PR-A-021 .....	148
TABLA 144. PRUEBA PR-A-022 .....	148
TABLA 145. PRUEBA PR-A-023 .....	149
TABLA 146. PRUEBA PR-A-024 .....	149
TABLA 147. PRUEBA PR-A-025 .....	150
TABLA 148. PRUEBA PR-A-026 .....	150
TABLA 149. PRUEBA PR-A-027 .....	150
TABLA 150. PRUEBA PR-A-028 .....	151
TABLA 151. PRUEBA PR-A-029 .....	151
TABLA 152. PRUEBA PR-A-030 .....	151
TABLA 153. PRUEBA PR-A-031 .....	152
TABLA 154. PRUEBA PR-A-032 .....	152
TABLA 155. PRUEBA PR-A-033 .....	153
TABLA 156. PRUEBA PR-A-034 .....	153
TABLA 157. PRUEBA PR-A-035 .....	153
TABLA 158. PRUEBA PR-A-036 .....	154

---

---



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

TABLA 159. PRUEBA PR-A-037 .....	154
TABLA 160. PRUEBA PR-A-038 .....	154
TABLA 161. PRUEBA PR-A-039 .....	155
TABLA 162. PRUEBA PR-A-040 .....	155
TABLA 163. PRUEBA PR-A-041 .....	155
TABLA 164. PRUEBA PR-A-042 .....	156
TABLA 165. PRUEBA PR-A-043 .....	156
TABLA 166. PRUEBA PR-A-044 .....	157
TABLA 167. PRUEBA PR-A-045 .....	157
TABLA 168. PRUEBA PR-A-046 .....	158
TABLA 169. PRUEBA PR-A-047 .....	158
TABLA 170. PRUEBA PR-A-048 .....	159
TABLA 171. PRUEBA PR-A-049 .....	160
TABLA 172. PRUEBA PR-A-050 .....	161
TABLA 173. PRUEBA PR-A-051 .....	162
TABLA 174. PRUEBA PR-A-052 .....	162
TABLA 175. PRUEBA PR-A-053 .....	163
TABLA 176. PRUEBA PR-A-054 .....	163
TABLA 177. PRUEBA PR-A-055 .....	163
TABLA 178. PRUEBA PR-A-056 .....	164
TABLA 179. PRUEBA PR-A-057 .....	164
TABLA 180. MATRIZ DE TRAZABILIDAD PRUEBAS ACEPTACIÓN – REQUISITOS DE CAPACIDAD .....	166
TABLA 181. PRUEBA PR-I-001.....	167
TABLA 182. PRUEBA PR-I-002.....	168
TABLA 183. PLAN DE PRUEBAS A DE DOMINIO A .....	169
TABLA 184. PLAN DE PRUEBAS B DE DOMINIO A .....	170
TABLA 185. PLAN DE PRUEBAS A DE DOMINIO B .....	172
TABLA 186. PLAN DE PRUEBAS B DE DOMINIO B.....	172
TABLA 187. PLANTILLA DE INFORME DE PRUEBAS .....	173

---

---



---

## Índice de Figuras

FIGURA 1. DIAGRAMA DE CASOS DE USO DE LA APLICACIÓN.....	15
FIGURA 2. CICLO DE VIDA. ....	41
FIGURA 3. RBS. ....	45
FIGURA 4. PROCESO WBS - ESTUDIO DE VIABILIDAD .....	46
FIGURA 5. PROCESO WBS - GESTIÓN DEL PROYECTO .....	46
FIGURA 6. PROCESO WBS - PRIMER PROTOTIPO .....	47
FIGURA 7. PROCESO WBS - SEGUNDO PROTOTIPO.....	48
FIGURA 8. PROCESO WBS - TERCER PROTOTIPO.....	49
FIGURA 9. PROCESO WBS MANUAL DE USUARIO .....	50
FIGURA 10. DIAGRAMA DE GANTT (GENERAL) .....	50
FIGURA 11. DIAGRAMA DE GANTT (ESTUDIO DE VIABILIDAD) .....	51
FIGURA 12. DIAGRAMA DE GANTT (GESTIÓN DEL PROYECTO) .....	52
FIGURA 13. DIAGRAMA DE GANTT (PRIMER PROTOTIPO) .....	53
FIGURA 14. DIAGRAMA DE GANTT (SEGUNDO PROTOTIPO) .....	54
FIGURA 15. DIAGRAMA DE GANTT (TERCER PROTOTIPO) .....	55
FIGURA 16. DIAGRAMA DE GANTT (MANUAL DE USUARIO) .....	55
FIGURA 17. DIAGRAMA DE GANTT (ENTREGA) .....	56
FIGURA 18. DIAGRAMA DE SECUENCIA CASO DE USO CU-006 .....	85
FIGURA 19. DIAGRAMA DE SECUENCIA CASO DE USO CU-014 .....	86
FIGURA 20. DIAGRAMA CONCEPTUAL DE CLASES .....	87
FIGURA 21. INTERFAZ TRATAMIENTO .....	97
FIGURA 22. INTERFAZ DE IMPORTACIÓN .....	99
FIGURA 23. INTERFAZ DE ENTRENAMIENTO .....	100
FIGURA 24. INTERFAZ GESTOR DE LOS FICHEROS DE RED .....	101
FIGURA 25. DIAGRAMA DE SECUENCIA EXTENDIDO DEL CASO DE USCO CU-006 .....	116
FIGURA 26. DIAGRAMA DE SECUENCIA EXTENDIDO DEL CASO DE USO CU-014 .....	117
FIGURA 27. DIAGRAMA DE CLASES DE DISEÑO.....	119
FIGURA 28. EJEMPLO DE FICHERO DE CONFIGURACIÓN .....	130
FIGURA 29. INTERFAZ TRATAMIENTO .....	174
FIGURA 30. INTERFAZ ENTRENAMIENTO.....	175
FIGURA 31. FICHERO RESUMEN PLAN DE PRUEBAS A DEL DOMINIO A.....	177
FIGURA 32. INFLUENCIA DEL FICHERO RESUMEN PLAN DE PRUEBAS B DEL DOMINIO A .....	177
FIGURA 33. FICHERO RESUMEN PLAN DE PRUEBAS B DEL DOMINIO A.....	178
FIGURA 34. INFLUENCIA DEL FICHERO RESUMEN PLAN DE PRUEBAS B DEL DOMINIO A .....	178
FIGURA 35. FICHERO RESUMEN PLAN DE PRUEBAS A DEL DOMINIO B.....	179
FIGURA 36. FICHERO RESUMEN PLAN DE PRUEBAS A DEL DOMINIO B.....	180
FIGURA 37. ARRANQUE APLICACIÓN.....	184
FIGURA 38. INTERFAZ TRATAMIENTO .....	185

---



---

FIGURA 39. SELECCIONAR UN FICHERO DE ENTRADA .....	186
FIGURA 40. SELECCIÓN RUTA DE SALIDA MANUALMENTE.....	187
FIGURA 41. SELECCIÓN RUTA DE SALIDA .....	188
FIGURA 42. IMPORTAR DATOS .....	189
FIGURA 43. SELECCIONAR ATRIBUTOS .....	190
FIGURA 44. ATRIBUTOS SELECCIONADOS.....	191
FIGURA 45. SELECCIÓN ATRIBUTOS DE SALIDA .....	192
FIGURA 46. SELECCIONAR TÉCNICA DE DIVISIÓN .....	194
FIGURA 47. NÚMERO DE PRUEBAS.....	195
FIGURA 48. NÚMERO DE CONJUNTOS .....	196
FIGURA 49. CONJUNTOS DE INSTANCIAS .....	197
FIGURA 50. REPARTO DE INSTANCIAS TOTALMENTE ALEATORIO .....	198
FIGURA 51. REPARTO DE INSTANCIAS DIRIGIDO .....	199
FIGURA 52. AÑADIR EXPERIMENTO.....	200
FIGURA 53. AÑADIR EXPERIMENTO DE FORMA AUTOMATIZADA .....	201
FIGURA 54.RESULTADO AÑADIR EXPERIMENTOS DE FORMA AUTOMATIZADA.....	202
FIGURA 55. ELIMINAR VARIOS EXPERIMENTOS .....	203
FIGURA 56. RESULTADO DE EXPERIMENTOS ELIMINADOS .....	204
FIGURA 57. GENERACIÓN DE FICHEROS.....	205
FIGURA 58. RESULTADO GENERACIÓN FICHEROS .....	206
FIGURA 59. FICHEROS GENERADOS .....	207
FIGURA 60. GUARDAR CONFIGURACIÓN.....	208
FIGURA 61.RESULTADO GUARDAR CONFIGURACIÓN.....	209
FIGURA 62. CARGAR CONFIGURACIÓN .....	210
FIGURA 63. RESULTADO CARGAR CONFIGURACIÓN .....	211
FIGURA 64. MOSTRAR INTERFAZ ENTRENAMIENTO.....	212
FIGURA 65. INTERFAZ DE ENTRENAMIENTO .....	213
FIGURA 66. SELECCIÓN RUTA FICHEROS DE RED .....	214
FIGURA 67. IDENTIFICACIÓN FICHEROS DE RED.....	215
FIGURA 68. SELECCIÓN DE TIPOLOGÍA .....	216
FIGURA 69. RESULTADO CAMBIAR MANUALMENTE TIPOLOGÍA DE RED .....	217
FIGURA 70. CRITERIOS DE PARADA .....	219
FIGURA 71. FICHERO EVALUACIÓN Y RESUMEN .....	220
FIGURA 72. ESTABLECER RUTA FICHERO RESUMEN.....	221
FIGURA 73. SELECCIONAR ELEMENTOS ENTRENAMIENTO .....	223
FIGURA 74. ESTADO ENTRENAMIENTO.....	223
FIGURA 75. GENERACIÓN DE FICHEROS DE EVALUACIÓN Y RESUMEN.....	224
FIGURA 76. ABRIR INFORME DE EVALUACIÓN.....	225
FIGURA 77. INFORME EVALUACIÓN ENTRENAMIENTO CLASIFICACIÓN .....	226
FIGURA 78. INFORME EVALUACIÓN ENTRENAMIENTO REGRESIÓN.....	227

---



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

FIGURA 79. ABRIR FICHERO INFORME DE RESUMEN. ....	228
FIGURA 80. INFORME RESUMEN ENTRENAMIENTO CLASIFICACIÓN.....	229
FIGURA 81. INFORME RESUMEN ENTRENAMIENTO REGRESIÓN .....	229
FIGURA 82. SALIR DE LA APLICACIÓN .....	230
FIGURA 83. APLICACIÓN NEUROSOLUTIONS .....	232
FIGURA 84. ELECCIÓN TIPOLOGÍA RED EN NEURAL BUILDER .....	233
FIGURA 85. INTRODUCCIÓN FICHERO DE ENTRENAMIENTO .....	234
FIGURA 86. FICHERO TRAIN .....	235
FIGURA 87. PARÁMETROS FICHERO TRAIN .....	236
FIGURA 88. SELECCIÓN FICHERO ENTRENAMIENTO Y VALIDACIÓN CRUZADA.....	237
FIGURA 89. NÚMERO CAPAS OCULTAS .....	238
FIGURA 90. NÚMERO DE NEURONAS OCULTAS .....	239
FIGURA 91. CONFIGURACIÓN CAPA DE SALIDA .....	240
FIGURA 92 CONFIGURACIÓN CONTROL DE APRENDIZAJE SUPERVISADO.....	241
FIGURA 93. CONFIGURAR SALIDA .....	242
FIGURA 94. EJEMPLO RED NEUROSOLUTIONS .....	243
FIGURA 95. GUARDAR RED DE NEUROSOLUTIONS .....	244

---

---



# 1 Introducción

El presente proyecto surge de la necesidad de crear una aplicación que permita la automatización del proceso de aprendizaje de redes neuronales artificiales [1]. Esta automatización facilita al sistema la capacidad de crear, desarrollar y evaluar series de experimentos que pueden ejecutarse de manera secuencial e iterativa. Este hecho además facilita el aprendizaje y posterior testeo de las redes neuronales que se van a manejar a través de la aplicación, ya que libera al usuario de la carga de realizar experimento a experimento las pruebas de forma manual.

Debido a las múltiples aplicaciones que se pueden dar a la simulación de datos en redes neuronales, el presente proyecto pretende que puede aplicarse a determinados conjuntos de datos bajo un dominio. De este modo se podrá proporcionar al usuario una aplicación que permita realizar simulaciones en diversos dominios de investigación.

El proyecto contempla también el procesamiento previo de los datos de entrada para el sistema, permitiendo la preparación de diversos experimentos para un dominio dado. Esta funcionalidad se integra en el proyecto actual desde un proyecto anterior [11].

Por último, para facilitar la interacción del usuario con la aplicación a desarrollar, se pretende dotar a éstos de interfaces intuitivas y fáciles de manejar, así como de manuales de usuario para su correcta utilización.

Los objetivos que se pretenden alcanzar mediante la realización de este proyecto se exponen a continuación:

- Crear un sistema capaz de cubrir todas las necesidades planteadas por el cliente.
- Seguir la metodología de desarrollo Métrica Versión 3 con el fin de generar software de calidad. En este caso, debido a las características particulares del proyecto se seguirá una adaptación de dicha metodología.



- Respetar la planificación establecida al principio del proyecto, salvo ligeras modificaciones, para poder cumplir los plazos de entrega establecidos.
- Poner en práctica los conocimientos adquiridos durante la realización de la carrera, ampliando los conocimientos acerca de los diferentes roles que intervienen en el desarrollo de un proyecto.
- Proporcionar toda la información necesaria para la utilización, mantenimiento y ampliación de la aplicación.

El presente documento recopila toda la información empleada durante el desarrollo del proyecto. Contiene todos los documentos generados en las distintas etapas del mismo, siguiendo la siguiente estructura:

- **Introducción:** Explica cómo surge el proyecto, cuáles son los objetivos a alcanzar con su realización, y la estructura de la documentación presentada.
- **Estudio de la Viabilidad del Sistema:** Realiza un estudio de la solicitud del cliente y de la situación actual, proponiendo una solución inicial al problema planteado por el cliente en base a las restricciones económicas, técnicas, legales y operativas.
- **Gestión del Proyecto:** Detalla todos los aspectos relacionados con la gestión del proyecto, tales como los recursos necesarios, una planificación detallada, y una estimación de los costes que supone su realización.
- **Análisis del Sistema:** Recoge el conjunto de requisitos que ha de cumplir el sistema a construir.
- **Diseño del Sistema:** Define el diseño del sistema de forma exhaustiva y con un nivel de detalle profundo. También realiza un estudio de toda la tecnología útil para la realización del sistema.
- **Plan de Validación y Verificación del Software:** Ofrece un catálogo de pruebas que servirán para comprobar el correcto funcionamiento del sistema y para verificar el cumplimiento de las funcionalidades requeridas por el cliente.





## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

---

- **Construcción del sistema de información:** Detalla la fase de codificación el sistema desarrollado.
- **Conclusiones:** Resume las conclusiones alcanzadas tras la realización del presente proyecto.
- **Anexos:** Incluye un documento con el formato recomendado para los ficheros de datos de entrada, así como los manuales de usuario para el sistema desarrollado.



## 2 Definiciones y acrónimos

### 2.1 Definiciones

- **Backup:** Copia de seguridad.
- **Clasificación:** Predicción de la clase a la que pertenece un determinado dato a partir de unos datos de entrada.
- **Componente:** Elemento software de un sistema informático que internamente se estructura en clases.
- **Entorno de desarrollo:** Aplicación compuesta por un conjunto de herramientas para la programación de aplicaciones.
- **Entrenamiento:** Proceso al cual se somete a una red de neuronas artificial para que adquiriera conocimiento a partir de unos datos.
- **Epoch:** Un paso en el proceso de formación de una red neuronal artificial
- **Excel:** Aplicación de tipo hoja de cálculo incluida en la suite ofimática Microsoft Office.
- **Experimento:** Conjunto de datos preparados para el procesamiento de una red de neuronas artificial.
- **Extensión (de fichero):** Identificador de 3 caracteres situado a continuación del nombre de un fichero que permite definir el tipo del propio fichero.
- **Factor de correlación:** El factor de correlación se calcula cuando se realiza la predicción de una variable numérica, se obtiene el factor de correlación medio para cada una de las variables a predecir. Cuanto más se aproxime este valor a 1 más fuerte es la aproximación entre la predicción de la red neuronal y el valor real de la variable.
- **Hoja de datos:** Elemento de un documento de la aplicación Microsoft Excel que contiene la información estructurada en forma de hoja de cálculo.



- **Influencia:** Se representa mediante un valor que oscila entre 0 y 1, siendo la relación que hay entre el número de veces que la variable estaba fuera de rango cuando el test ha fallado, y el número total de veces que la variable ha estado fuera de rango
- **Instancia:** Dato acerca de un dominio compuesto por una serie de atributos con sus correspondientes valores.
- **Jordan-Elman:** La red de Jordan-Elman es un tipo de red neuronal. Forma parte de las denominadas redes parcialmente recurrentes. Tienen la forma general de un Multilayer Perceptron, pero se añaden conexiones recurrentes. Está constituida por una capa oculta con función de activación tangente hiperbólica y una capa de salida con función de activación lineal. Las conexiones recurrentes parten de la salida de la capa oculta y se realimentan como entradas a esta misma capa. Las recurrencias de este modelo neuronal le permite tanto detectar como generar patrones que van cambiando a lo largo del tiempo.
- **Kernel:** Un Kernel es una función de ponderación.
- **Lenguaje de programación:** Lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones.
- **Localización:** Ruta de acceso a un fichero o directorio del ordenador.
- **MEAN Square Error:** el error cuadrado medio de un estimador es una de las muchas maneras de cuantificar la diferencia entre un estimador y el verdadero valor de la cantidad que se estima.
- **MSE :** ver. *MEAN Square Error*
- **Multilayer Perceptron:** El perceptrón multicapa es un tipo de red neuronal. Todas las neuronas de una capa están conectadas a las neuronas de la capa precedente. Todas las señales fluyen en la misma dirección: hacia adelante. La capa de entrada generalmente no está constituida por neuronas, ya que no realizan procesamiento alguno,

siendo su única función la de alimentar el vector de entrada a la primera capa oculta.

- **Prototipo:** Elemento tomado como modelo sobre el que trabajar posteriormente.
- **Radial basis function:** Las redes neuronales de Funciones Base Radiales es un tipo de red neuronal. Son del tipo alimentadas hacia delante, con una arquitectura de tres capas. La capa de entrada se utiliza para presentar a la red los patrones de entrada. La segunda capa o capa oculta está constituida por neuronas no lineales, y la capa de salida por neuronales con funciones de transferencia lineales.
- **Recurrent Network:** Una red de recurrente es un tipo de red neuronal. Tiene una única capa de neuronas, donde cada neurona se conecta con todas las demás, configurando una red recurrente completamente conectada
- **Red de Neuronas Artificial:** Simulación informática que trata de imitar el comportamiento de las redes de neuronas biológicas para adquirir conocimiento.
- **Regresión:** Predicción del valor numérico de una variable.
- **Self-organizing feature map:** Los mapas autoorganizados son un tipo de red neuronal no supervisada, competitiva, distribuida de forma regular en una rejilla de, normalmente, dos dimensiones, cuyo fin es descubrir la estructura subyacente de los datos introducidos en ella.
- **Stakeholders:** Aquellas personas que afectan o son afectados por el desarrollo del proyecto.
- **Support Vector Machine:** Una red de máquinas de soporte vectorial es un tipo de red neuronal. Está basada en kernels que realiza clasificación lineal sobre vectores transformados a un espacio de dimensión superior, es decir, separa mediante un hiperplano en el espacio transformado
- **Test:** Ver *Validación*.



- **Texto plano:** Se refiere a ficheros compuestos únicamente de texto sin formato alguno.
- **Train:** Ver *Entrenamiento*.
- **Validación:** Proceso al que se somete a una red de neuronas para comprobar si el entrenamiento ha sido correcto y la red ha adquirido el conocimiento esperado.
- **Validación cruzada:** Mecanismo utilizado durante el proceso de entrenamiento de una red de neuronas para evitar el sobreaprendizaje de la misma.



## **2.2 Acrónimos**

- **CU:** Caso de uso
- **RS:** Requisito Software
- **RU-C:** Requisito de Usuario de Capacidad
- **RU-R:** Requisito de Usuario de Restricción



## **3 Estudio de Viabilidad del Sistema**

### **3.1 Introducción**

#### **3.1.1 Propósito del plan**

El objetivo del Estudio de Viabilidad del Sistema es realizar un análisis detallado de las necesidades del cliente, con el fin de proponer una solución en un plazo establecido. Este estudio debe tener en cuenta restricciones de diversa índole: económicas, técnicas, legales y operativas.

Basándose en las necesidades del cliente, se detallará el alcance del sistema, es decir, se estudiará el alcance de la necesidad planteada por el cliente identificando los primeros requisitos, estructuras implicadas, suposiciones y restricciones así como los *stakeholders*, el equipo de trabajo y la planificación a seguir por éste.

Una vez definido el alcance del sistema se realizará un estudio de la situación actual de los sistemas similares existentes en la actualidad. El objetivo de esta tarea consiste en identificar los recursos de información existentes, posibles problemas y mejoras.

A continuación se realizará una definición de los requisitos que debe cumplir el sistema, de manera que se obtengan un conjunto de necesidades detalladas, no ambiguas, y completas, que sirva de base para las siguientes etapas del ciclo de vida del proyecto. Dichos requisitos se describirán en un lenguaje sencillo con el propósito de que el cliente los comprenda y pueda validarlos sin problemas.

Finalmente se expondrá la solución elegida para ser desarrollada por el equipo de trabajo.



## **3.2 *Establecimiento del alcance del sistema***

El objetivo de esta actividad es realizar una primera aproximación al estudio de los requisitos, identificar las unidades organizativas afectadas y analizar las suposiciones y restricciones del sistema. También se identificarán a las personas que deben participar en el estudio de viabilidad, así como sus perfiles, especificando sus tareas y responsabilidades dentro del proceso.

### **3.2.1 Estudio de la solicitud**

En la solicitud del cliente se especifica la necesidad de desarrollar una aplicación capaz de realizar el tratamiento de ficheros de entrada, preparando todos los ficheros de datos para los experimentos que se deseen realizar, automatizar el proceso de entrenamiento de varias redes neuronales artificiales y evaluar los resultados conseguidos.

Con el desarrollo de este proyecto se pretende desarrollar una aplicación que cumpla las siguientes características:

- Interaccionar con una aplicación de redes de neuronas artificiales.
- Generar informes de evaluación.
- Controlar errores en la introducción de parámetros de entrada por parte del usuario.
- Entrenamiento de experimentos en varias tipologías de redes neuronales.
- Permitir la entrada de datos multidominio.

Tras el estudio de la solicitud del cliente, el equipo de desarrollo del proyecto ha determinado que el cliente propone un proyecto perfectamente viable.

La persona encargada de llevar a cabo este proyecto es Israel González Carrasco, como directores del mismo, y Miguel Fortes Gómez, como desarrollador.





### **3.2.2 Identificación del alcance del sistema**

El presente proyecto comprende una aplicación a desarrollar que permita el tratamiento de ficheros para los experimentos que se deseen realizar, automatizar el proceso de entrenamiento de varias redes neuronales artificiales y evaluar los resultados conseguidos.

En este apartado se pretenden identificar los requisitos de cada una de las partes, identificando las principales funcionalidades que se deberán desarrollar.

En primer lugar, sobre tratamiento de ficheros de entrada debe proporcionar al usuario la funcionalidad principal de generar los ficheros necesarios para la posterior experimentación con el módulo de entrenamiento. Estos ficheros se deben generar atendiendo a una serie de parámetros a decidir por el usuario, tales como:

- Técnica de división de instancias.
- Tipo de reparto entre conjuntos de entrenamiento, validación y validación cruzada.
- Selección de atributos a importar del fichero de datos original.
- Número de pruebas del experimento.

Merece la pena destacar que la aplicación debe ser capaz de tratar los ficheros de datos originales en formato de Microsoft Excel, mientras que los generados por la propia aplicación deben ser en formato de texto plano.

En segundo lugar, sobre la automatización del proceso de entrenamiento de varias redes neuronales artificiales, se pretende reducir el tiempo de proceso al no tener que utilizar un sistema real.

Por último, se debe proporcionar la funcionalidad de generar informes de evaluación que permitan al usuario analizar en profundidad los resultados obtenidos en sus experimentos. Dichos informes deberán ser realizados en formato de Microsoft Excel.



### **3.2.3 Identificación de los interesados en el sistema (stakeholders)**

En este punto se lleva a cabo una identificación de todas aquellas personas o entidades interesadas de alguna forma en el proyecto. Se describen a continuación:

- **Cliente:** Es la persona que realiza la solicitud de desarrollo del sistema, en este caso, el cliente es Israel González Carrasco, profesor de la Universidad Carlos III de Madrid.
- **Usuarios gestores:** Son aquellas personas que harán uso de la aplicación.
- **Tutor y coordinador del proyecto:** Israel González Carrasco
- **Autor del Proyecto:** Miguel Fortes Gómez. Es la persona encargada de la realización del proyecto, siendo además el máximo responsable del mismo y de todos los productos generados durante su desarrollo.



### **3.3 *Estudio de la situación actual***

La situación actual es el estado en el que se encuentran los sistemas de información existentes en el momento en el que se inicia su estudio.

#### **3.3.1 Valoración del estudio de la situación actual**

En la actualidad, las estrategias y modelos empleados para el diseño de aplicaciones basados en redes de neuronas se caracterizan por la resolución de problemas en un ámbito determinado usando una tipología de red determinada [3] [4] [5].

#### **3.3.2 Realización del diagnóstico actual**

Una vez realizado el estudio de las técnicas y herramientas existentes en la actualidad, se puede concluir que no existe un sistema que proporcione la misma funcionalidad que se pretende alcanzar con este proyecto.

Mientras que la mayoría de sistemas y aplicaciones existentes se han desarrollado para la resolución de un problema concreto, con el presente proyecto se pretende desarrollar la arquitectura para la resolución de problemas de cualquier dominio sobre el que se tengan datos suficientes.



### **3.4 Definición de los requisitos del sistema**

Una vez realizado el estudio de la solicitud del cliente se realiza la extracción de requisitos con el fin de cumplir con las funcionalidades debe proporcionar la aplicación.

Esta extracción de requisitos se realiza también con el objetivo de orientar al cliente, permitiendo al equipo de desarrollo obtener una lista de requisitos detallada, completa y sin ambigüedades. Estos requisitos proporcionarán una visión general de la aplicación, sin ahondar en aspectos técnicos, estableciendo las principales funcionalidades y restricciones, sirviendo de base a posteriores procesos del ciclo de vida.

Previamente a la redacción de los requisitos han sido identificados los casos de uso con el fin de lograr una mejor definición.

### 3.4.1 Especificación de los casos de uso

A continuación se muestran los diagramas de casos de uso para cada uno de los módulos del sistema que se van a desarrollar en este proyecto:

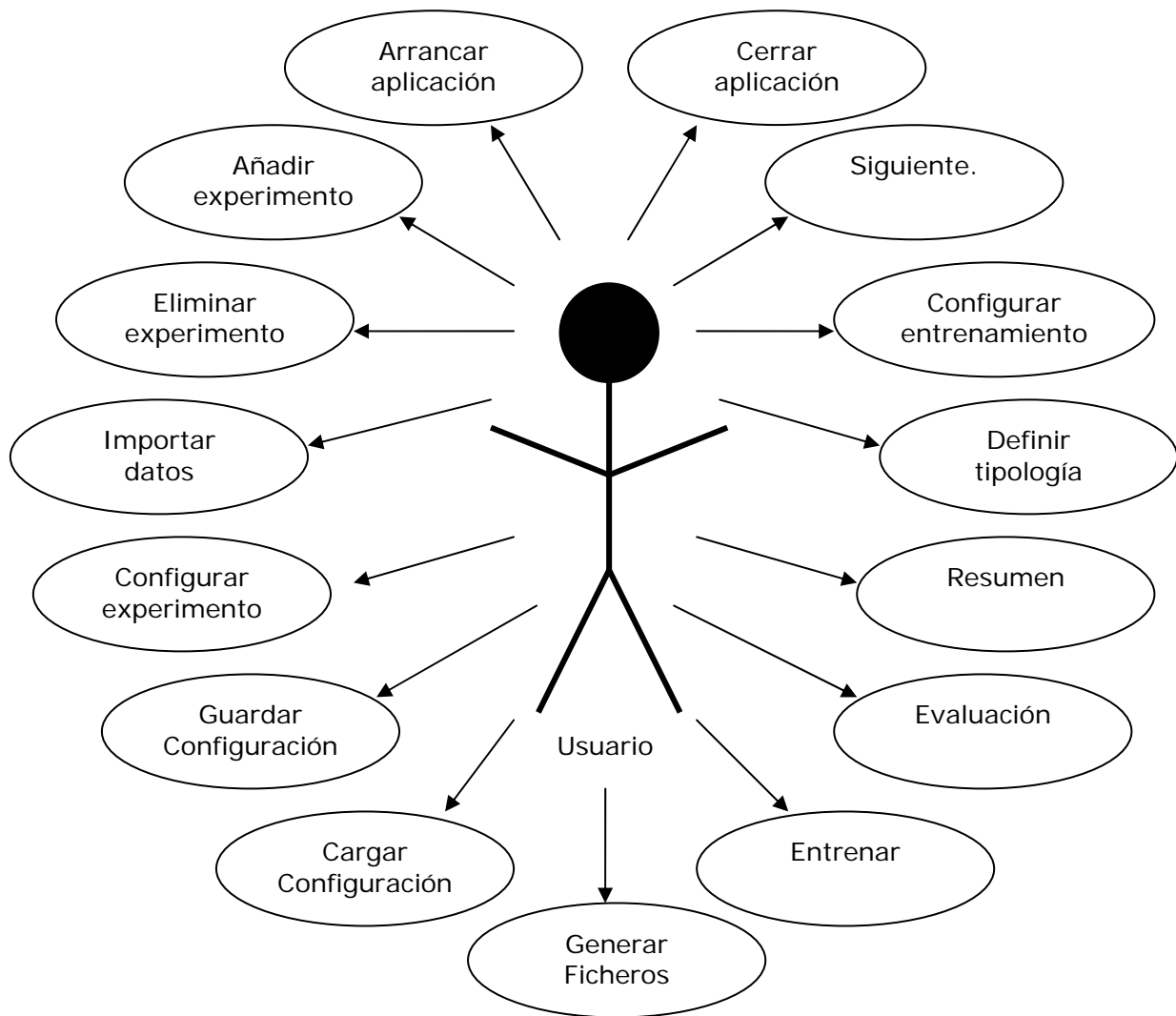


Figura 1. Diagrama de Casos de Uso de la aplicación



### **3.4.1.1 Especificación detallada de los casos de uso**

En este apartado se realiza la especificación de los casos de uso. Cada caso de uso estará especificado por los siguientes atributos:

- **Identificador:** Identifica al caso de uso de forma única. Debe seguir el formato: *CU- XXX*, siendo XXX un valor numérico único para cada caso de uso.
- **Nombre:** Breve especificación textual del caso de uso.
- **Actores:** Tipo de usuario del sistema que inicia el caso de uso.
- **Objetivo:** Finalidad del caso de uso.
- **Precondiciones:** Estado previo que se debe cumplir para poder realizar una operación.
- **Postcondiciones:** Estado en el que queda el sistema tras realizar una operación.
- **Escenario básico:** Especifica la manera en la que interactúa un actor con el sistema y cuál es la respuesta que el sistema le ofrece.
- **Escenarios alternativos:** Condiciones excepcionales que afectan al escenario y respuestas del sistema ante esas situaciones.



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	CU-001
Nombre	Arrancar aplicación.
Actores	Usuario.
Objetivo	Inicia la aplicación de forma correcta.
Precondiciones	Ninguna.
Poscondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li></ul>
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario inicia la ejecución de la aplicación.</li><li>2. La aplicación muestra la interfaz de tratamiento de usuario.</li></ol>
Escenarios alternativos	

Tabla 1. Caso de uso CU-001

Identificador	CU-002
Nombre	Importar Datos.
Actores	Usuario.
Objetivo	Importar datos desde el fichero de entrada en formato Microsoft Excel a la aplicación para generar los ficheros de los experimentos.
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li></ul>
Poscondiciones	Datos importados.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario selecciona un fichero de entrada.</li><li>2. La aplicación muestra las hojas de datos disponibles.</li><li>3. El usuario selecciona una hoja de datos y hace clic sobre "Importar datos".</li><li>4. La aplicación muestra los atributos disponibles.</li><li>5. El usuario selecciona los atributos que desee y hace clic sobre "Importar".</li><li>6. La aplicación muestra los atributos importados en la lista "Atributos del fichero de datos".</li></ol>
Escenarios alternativos	

Tabla 2. Caso de uso CU-002



Identificador	CU-003
Nombre	Configurar experimento.
Actores	Usuario.
Objetivo	Configurar los parámetros de un experimento antes de añadirlo a la lista de "Experimentos a preparar".
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li></ul>
Poscondiciones	Experimento configurado.
Escenario básico	1. El usuario configura las opciones del experimento a través de la interfaz de tratamiento de usuario.
Escenarios alternativos	

Tabla 3. Caso de uso CU-003

Identificador	CU-004
Nombre	Añadir experimento.
Actores	Usuario.
Objetivo	Añadir un experimento configurado a la lista de "Experimentos a preparar".
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li><li>• Experimento configurado.</li></ul>
Poscondiciones	Experimento añadido a la lista de "Experimentos a preparar".
Escenario básico	1. El usuario hace clic sobre el botón "Añadir experimento(s)". 2. La aplicación muestra el experimento añadido en la lista "Experimentos a preparar" y limpia los campos de configuración.
Escenarios alternativos	

Tabla 4. Caso de uso CU-004





## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	CU-005
Nombre	Eliminar experimentos.
Actores	Usuario.
Objetivo	Eliminar experimentos de la lista de "Experimentos a generar".
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li><li>• Al menos un experimento añadido en la lista "Experimentos a preparar".</li></ul>
Poscondiciones	Experimentos eliminados.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario selecciona los experimentos de la lista "Experimentos a preparar" que desee eliminar.</li><li>2. El usuario hace clic sobre el botón "Eliminar experimento(s)".</li><li>3. La aplicación elimina los experimentos seleccionados de la lista "Experimentos a eliminar" y muestra un mensaje informativo al usuario.</li></ol>
Escenarios alternativos	

Tabla 5. Caso de uso CU-005

Identificador	CU-006
Nombre	Generar ficheros.
Actores	Usuario.
Objetivo	Generar los ficheros correspondientes a los experimentos seleccionados añadidos a la lista "Experimentos a preparar".
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li><li>• Al menos un experimento añadido a la lista "Experimentos a preparar".</li></ul>
Poscondiciones	Ficheros de experimentación generados.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario hace clic sobre el botón "Generar Seleccionados".</li><li>2. La aplicación genera los ficheros correspondientes a cada experimento seleccionado de la lista "Experimentos a preparar".</li><li>3. La aplicación muestra un mensaje informativo al usuario.</li></ol>
Escenarios alternativos	

Tabla 6. Caso de uso CU-006



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	CU-007
Nombre	Guardar configuración.
Actores	Usuario.
Objetivo	Guardar la configuración de la aplicación.
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li><li>• Al menos un experimento configurado y añadido a la lista "Experimentos a preparar".</li></ul>
Poscondiciones	Configuración guardada.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario hace clic sobre el botón "Guardar configuración".</li><li>2. La aplicación almacena la configuración existente en ese momento en un fichero en formato Excel.</li></ol>
Escenarios alternativos	

Tabla 7. Caso de uso CU-007

Identificador	CU-008
Nombre	Cargar configuración.
Actores	Usuario.
Objetivo	Cargar una configuración de la aplicación almacenada previamente.
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li><li>• Contar con un fichero de configuración almacenado previamente.</li></ul>
Poscondiciones	Configuración cargada.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario hace clic sobre el botón "Cargar configuración" y selecciona el fichero de configuración almacenado previamente.</li><li>2. La aplicación restaura la configuración atendiendo a la información almacenada en el fichero.</li></ol>
Escenarios alternativos	

Tabla 8. Caso de uso CU-008



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	CU-009
Nombre	Cerrar aplicación.
Actores	Usuario.
Objetivo	Cerrar la aplicación de forma correcta.
Precondiciones	Aplicación arrancada.
Poscondiciones	Aplicación cerrada.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario hace clic sobre el botón "Salir".</li><li>2. La aplicación se cierra completamente.</li></ol>
Escenarios alternativos	

Tabla 9. Caso de uso CU-009

Identificador	CU-010
Nombre	Siguiente.
Actores	Usuario.
Objetivo	Mostrar la interfaz de entrenamiento de usuario
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de tratamiento mostrada.</li><li>• Ficheros de Experimentación generados.</li></ul>
Poscondiciones	Interfaz de entrenamiento mostrada.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario hace clic sobre el botón "Siguiente".</li><li>2. La aplicación muestra la interfaz de entrenamiento y oculta la interfaz tratamiento.</li></ol>
Escenarios alternativos	

Tabla 10. Caso de uso CU-010



Identificador	CU-011
Nombre	Configurar Entrenamiento.
Actores	Usuario.
Objetivo	Configurar los parámetros y opciones del Entrenamiento.
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de entrenamiento mostrada.</li></ul>
Poscondiciones	Parámetros y opciones del Entrenamiento configurados.
Escenario básico	1. El usuario configura las opciones del entrenamiento a través de la interfaz de entrenamiento de usuario.
Escenarios alternativos	

Tabla 11. Caso de uso CU-011

Identificador	CU-012
Nombre	Definir tipología.
Actores	Usuario.
Objetivo	Cambiar la tipología identificada por la aplicación
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de entrenamiento mostrada.</li></ul>
Poscondiciones	Tipología cambiada
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario selecciona una carpeta que contenga los ficheros de red.</li><li>2. La aplicación muestra los ficheros encontrados indicando la tipología de la red de cada uno de ellos.</li><li>3. El usuario selecciona un fichero de la lista.</li><li>4. El usuario pulsa el botón "Cambiar Tipología Red".</li><li>5. La aplicación muestra las tipologías disponibles.</li><li>6. El usuario selecciona una tipología y pulsa el botón "aceptar".</li><li>7. La aplicación actualiza la lista con los cambios realizados por el usuario.</li></ol>
Escenarios alternativos	

Tabla 12. Caso de uso CU-012



Identificador	CU-013
Nombre	Anterior.
Actores	Usuario.
Objetivo	Mostrar la interfaz de tratamiento de usuario
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de entrenamiento mostrada.</li></ul>
Poscondiciones	Interfaz de tratamiento mostrada.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario hace clic sobre el botón "Anterior".</li><li>2. La aplicación muestra la interfaz de tratamiento y oculta la interfaz entrenamiento.</li></ol>
Escenarios alternativos	

Tabla 13. Caso de uso CU-013

Identificador	CU-014
Nombre	Entrenar.
Actores	Usuario.
Objetivo	Entrenar los experimentos seleccionados en las redes seleccionadas.
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de entrenamiento mostrada.</li><li>• Parámetros y opciones del Entrenamiento configurados.</li></ul>
Poscondiciones	Ficheros de Entrenamiento generados.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario hace clic sobre el botón "Entrenar Seleccionados".</li><li>2. La aplicación se comunica con la aplicación externa para entrenar los experimentos seleccionados en las redes seleccionadas.</li><li>3. La aplicación muestra una interfaz que indica el progreso del entrenamiento.</li></ol>
Escenarios alternativos	

Tabla 14. Caso de uso CU-014



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	CU-015
Nombre	Evaluación.
Actores	Usuario.
Objetivo	Muestra la Evaluación de un experimento seleccionado.
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de entrenamiento mostrada.</li><li>• Ficheros de Entrenamiento generados.</li><li>• "Crear ficheros de Resumen y Evaluación" marcado</li></ul>
Poscondiciones	Fichero evaluación mostrado.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario selecciona un experimento.</li><li>2. El usuario hace clic sobre el botón "Abrir Fichero Evaluación".</li><li>3. La aplicación abre el fichero correspondiente, mostrándolo al usuario.</li></ol>
Escenarios alternativos	

Tabla 15. Caso de uso CU-015

Identificador	CU-016
Nombre	Resumen.
Actores	Usuario.
Objetivo	Muestra el resumen del entrenamiento.
Precondiciones	<ul style="list-style-type: none"><li>• Aplicación arrancada.</li><li>• Interfaz de entrenamiento mostrada.</li><li>• Ficheros de Entrenamiento generados.</li><li>• "Crear ficheros de Resumen y Evaluación" marcado</li></ul>
Poscondiciones	Fichero resumen mostrado.
Escenario básico	<ol style="list-style-type: none"><li>1. El usuario hace clic sobre el botón "Abrir Fichero Resumen".</li><li>2. La aplicación abre el fichero correspondiente, mostrándolo al usuario.</li></ol>
Escenarios alternativos	

Tabla 16. Caso de uso CU-016



### 3.4.2 Definición de los requisitos del sistema

En este apartado se realiza una extracción de requisitos del sistema con el fin de presentar las principales funcionalidades deseadas por el cliente, sirviendo de base a posteriores fases del ciclo de vida del proyecto. Los requisitos identificados proporcionarán al cliente una visión general de la aplicación, de forma completa y sin ambigüedades.

#### 3.4.2.1 Identificación de los requisitos

En esta tarea se realiza la obtención detallada de requisitos de usuario mediante sesiones de trabajo con el cliente. La lista de requisitos obtenida no es definitiva y podrá ser modificada durante el proyecto, añadiendo requisitos no contemplados, o bien eliminando o modificando requisitos incorrectos, de forma que se terminen cubriendo todas las funcionalidades expuestas por el cliente.

La definición de requisitos recoge lo que quiere el cliente y lo que necesita, englobando los requisitos obtenidos del usuario en dos grandes categorías:

- **Requisitos de capacidad:** Representan lo que necesitan los usuarios para resolver un problema o lograr un objetivo.
- **Requisitos de restricción:** Son las restricciones impuestas por los usuarios sobre cómo se debe resolver el problema o cómo se debe alcanzar el objetivo.

Cada requisito de usuario debe incluir en su definición una serie de atributos, los cuales proporcionen toda la información necesaria para su seguimiento posterior y su clasificación. Estos atributos se describen a continuación:

- **Identificador:** Cada requisito de usuario debe estar identificado de forma única. Este identificador tendrá el siguiente formato: *RU-C-nnn* o *RU-R-nnn*, donde:
  - RU: Indica que se trata de un requisito de usuario.
  - C: Indica que se trata de un requisito de usuario de capacidad.
  - R: Indica que se trata de un requisito de usuario de restricción.
  - nnn: Tomará valores numéricos dentro del rango 000-999.



- **Prioridad:** Se asignará una prioridad a cada requisito con el fin de poder realizar una planificación correcta durante fases posteriores. Su clasificación puede tomar los valores: alta, media y baja.
- **Necesidad:** Los requisitos clasificados como esenciales para el usuario no pueden ser eliminados, mientras que los demás requisitos estarán siempre sujetos a modificación en el caso de que exista una causa que lo justifique. Por lo tanto, se clasificarán descendientemente de acuerdo a su necesidad de la siguiente forma: esencial, deseable y opcional.
- **Claridad:** Identifica la falta o existencia de ambigüedad de un requisito, esto es, si puede ser interpretado de varias formas dependiendo del contexto. Los requisitos serán clasificados de acuerdo a su claridad de forma descendente de la siguiente forma: alta, media y baja.
- **Fuente:** Identifica el origen del requisito, que puede estar en el usuario, una fuente externa como un documento, o el propio equipo de desarrollo del proyecto.
- **Estabilidad:** Algunos requisitos pueden no estar sujetos a cambios durante el proyecto debido a su naturaleza, mientras que otros pueden estar sujetos a determinados cambios por el desarrollo de la etapa de diseño o los requisitos software. Los requisitos poco estables deberán ser observados con mayor cuidado durante el desarrollo del proyecto al ser susceptibles a cambios. Cada requisito se clasificará en: estable o inestable.
- **Verificabilidad:** Indica si el cumplimiento de un requisito en el sistema puede ser susceptible de comprobación, esto es, si se puede verificar que el requisito se ha incorporado en el diseño y que en el sistema se puede verificar su cumplimiento. La verificabilidad de cada requisito se clasificará en: alta, media y baja.





### 3.4.2.2 Requisitos de capacidad

A continuación se presentan los requisitos de capacidad, los cuales representan lo que necesita el usuario para resolver un problema o lograr un objetivo.

Identificador	RU-C-001		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Iniciar la aplicación de forma que el usuario no deba borrar información introducida previamente.		

Tabla 17. Requisito de usuario RU-C-001

Identificador	RU-C-002		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Introducir una localización para los ficheros de entrada y de salida.		

Tabla 18. Requisito de usuario RU-C-002



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-C-003		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Importar datos desde un fichero de Microsoft Excel a la aplicación.		

Tabla 19. Requisito de usuario RU-C-003

Identificador	RU-C-004		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Repartir las instancias disponibles entre los conjuntos de entrenamiento, validación y validación cruzada.		

Tabla 20. Requisito de usuario RU-C-004



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-C-005		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Utilizar una entre las siguientes técnicas de división: <ul style="list-style-type: none"><li>Validación cruzada k-fold (KFCV).</li><li>Validación cruzada leave-one-out (LOOCV).</li><li>Remuestreo bootstrap (RBP).</li><li>Variante de KFCV (VKFCV).</li></ul>		

Tabla 21. Requisito de usuario RU-C005

Identificador	RU-C-006		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Añadir experimentos a la lista de "Experimentos a preparar" con los parámetros de configuración seleccionados.		

Tabla 22. Requisito de usuario RU-C-006



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-C-007		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Añadir experimentos a la lista de "Experimentos a preparar" con los parámetros de configuración seleccionados, incrementando o decrementando de forma automática los valores deseados por el usuario.		

Tabla 23. Requisito de usuario RU-C-007

Identificador	RU-C-008		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Eliminar experimentos añadidos previamente a la lista de "Experimentos a preparar".		

Tabla 24. Requisito de usuario RU-C-008



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-C-009		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Gestionar la configuración de la aplicación, permitiendo al usuario guardar y cargar el estado de la aplicación en un momento dado.		

Tabla 25. Requisito de usuario RU-C-009

Identificador	RU-C-010		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Generar los archivos de entrenamiento, validación cruzada y validación para los experimentos añadidos a la lista "Experimentos a preparar".		

Tabla 26. Requisito de usuario RU-C-010

**Proyecto Fin de Carrera**

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-C-011		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Cerrar la aplicación mediante el botón "Salir" situado en la interfaz.		

Tabla 27. Requisito de usuario RU-C-011

Identificador	RU-C-012		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Introducir una localización para los ficheros de red		

Tabla 28. Requisito de usuario RU-C-012

**Proyecto Fin de Carrera**

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-C-013		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Cambiar la identificación de la tipología de la red de un fichero de red.		

Tabla 29. Requisito de usuario RU-C-013

Identificador	RU-C-014		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Seleccionar el criterio de parada para el entrenamiento		

Tabla 30. Requisito de usuario RU-C-014



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-C-015		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Interaccionar con una aplicación de redes de neuronas artificiales para realizar las siguientes funciones: <ul style="list-style-type: none"><li>• Entrenar una red o varias redes.</li><li>• Validar una red o varias redes</li><li>• Poner una red o varias redes en producción.</li></ul>		

Tabla 31. Requisito de usuario RU-C-015

Identificador	RU-C-016		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Generar informes de evaluación en formato Excel, con la información necesaria para analizar los resultados obtenidos en la validación de las redes de neuronas artificiales.		

Tabla 32. Requisito de usuario RU-C-016





## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-C-017		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Generar informes de Resumen en formato Excel, con la información necesaria para analizar los resultados obtenidos en la validación de las redes de neuronas artificiales.		

Tabla 33. Requisito de usuario RU-C-017

Identificador	RU-C-018		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	La aplicación debe en primer lugar realizar el tratamiento de ficheros y luego realizar el entrenamiento de la red.		

Tabla 34. Requisito de usuario RU-C-018



### 3.4.2.3 Requisitos de restricción

Son las restricciones impuestas acerca de cómo se debe resolver el problema o como se debe alcanzar el objetivo.

Identificador	RU-R-001		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Generar archivos en formato de texto plano y con extensión ".txt".		

Tabla 35. Requisito de usuario RU-R-001

Identificador	RU-R-002		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Disponer de una interfaz sencilla e intuitiva que facilite el uso de la aplicación.		

Tabla 36. Requisito de usuario RU-R-002

**Proyecto Fin de Carrera**

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-R-003		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Mostrar el progreso de la generación de archivos mediante una barra de progreso en la interfaz.		

Tabla 37. Requisito de usuario RU-R-003

Identificador	RU-R-004		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Almacenar los archivos de configuración en formato de Microsoft Excel.		

Tabla 38. Requisito de usuario RU-R-004

**Proyecto Fin de Carrera**

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RU-R-005		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Desarrollar la aplicación en Microsoft Visual Studio 2005.		

Tabla 39. Requisito de usuario RU-R-005

Identificador	RU-R-06		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Mostrar el progreso del entrenamiento de las redes mediante una ventana informativa.		

Tabla 40. Requisito de usuario RU-R-06



### **3.5 *Estudio de alternativas de solución***

Con el desarrollo de este proyecto se pretende desarrollar una aplicación capaz de automatizar el proceso de entrenamiento de varias redes neuronales artificiales y evaluar los resultados conseguidos. La solución ha sido impuesta por el cliente y no aplica realizar un estudio de diferentes alternativas.

Dado que la aplicación esta basada en anteriores proyectos de los que se parte como base [11], se empleará como lenguaje de programación Visual Basic.NET [2].

En cuanto al entorno de desarrollo se ha optado por Microsoft Visual Studio 2005, dada su amplia capacidad y las funcionalidades ofrecidas para el lenguaje de programación elegido.



### **3.6 *Valoración de las alternativas***

No aplica porque la solución viene dada.

### **3.7 *Selección de la solución***

No aplica porque la solución viene dada.

## 4 Gestión del Proyecto

### 4.1 Ciclo de vida

En primer lugar se definirá este concepto para seguidamente explicar que actividades se realizan en cada una de las fases que establece el modelo elegido para finalmente justificar la elección del ciclo de vida.

Se entiende por ciclo de vida software el conjunto de fases, procesos y actividades requeridas para ofertar, desarrollar, probar, integrar, explotar y mantener un producto software indicando como funciones principales la de por un lado determinar el orden de las fases y procesos y por otro el de establecer los criterios de transición para pasar de una fase a otra.

Para el presente proyecto el ciclo de vida elegido ha sido el modelo “ciclo de vida con prototipado” (también llamado ciclo de vida iterativo por prototipos), que es el que más se adapta a las necesidades de desarrollo en este caso, ya que el propio producto a lo largo de su desarrollo puede considerarse una sucesión de prototipos que progresan hasta llegar al estado deseado. En cada ciclo (prototipo), las especificaciones se van resolviendo paulatinamente, completando las del ciclo anterior y ganando “madurez” hasta que se alcanza el punto en el cuál se completa y puede dejarse de entrar en un nuevo ciclo.

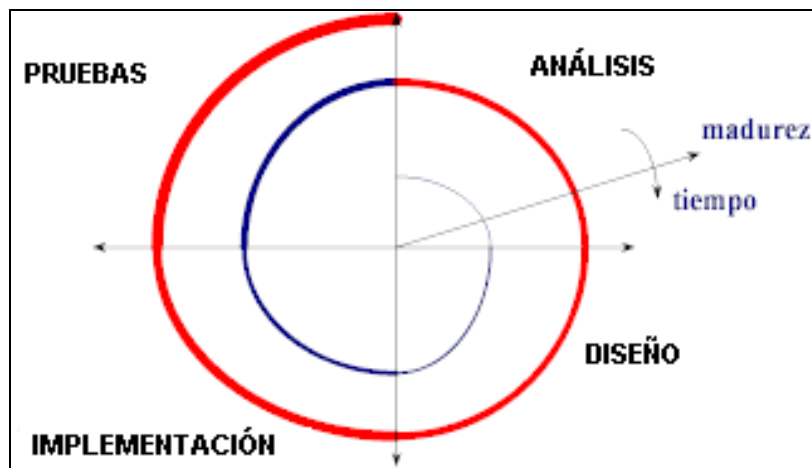


Figura 2. Ciclo de vida.



En cada ciclo se completan varias fases, que se explican brevemente a continuación:

- **Análisis:** En esta fase se determinan (o amplían) los requisitos software del sistema, especificando a un alto nivel la arquitectura de la solución que se propone para dichos requisitos.
- **Diseño:** En esta segunda fase, se diseñan tanto los interfaces de usuario de la aplicación, como la arquitectura a un nivel más bajo de especificación, detallando los procesos del sistema.
- **Implementación:** En esta etapa se codifican los interfaces de usuario, se codifican los procesos, y se documenta el manual de usuario.
- **Pruebas:** En esta última fase, se definen las pruebas a realizar por el prototipo en cuestión y se llevan a cabo.

Este ciclo ha sido considerado como el más adecuado debido a diferentes motivos:

- Permite evaluar distintas posibilidades de desarrollo, ya que en los prototipos se puede fácilmente añadir o eliminar funcionalidades.
- Permite al tutor o usuario final realizar un seguimiento periódico del proyecto para verificar si el alumno ha comprendido bien los requisitos y si está desarrollando la aplicación de acuerdo con éstos.
- El prototipo es un documento vivo de buen funcionamiento del producto final.
- Ayuda para determinar requerimientos expresados en el prototipo.
- Experimenta sobre los aspectos del sistema que representan mayor complejidad.
- El usuario reacciona mucho mejor ante el prototipo, sobre el que puede experimentar, que no sobre una especificación escrita.
- Se incrementa la calidad del producto final, ya que el prototipo permite trabajar.





Se debe realizar un buen estudio para determinar que este ciclo de vida es el necesario ya que si se comete algún error, se producirán los siguientes inconvenientes:

- Fuerte inversión en un producto que es desechable, los prototipos se descartan.
- Tendencia a tratar de convertir el prototipo mismo en el sistema de producción.
- Aumento del coste.
- Se arrastran decisiones del diseño de prototipos al producto final.



## **4.2 ORGANIZACIÓN DEL PROYECTO**

A la hora de llevar a cabo cualquier proyecto se hace necesaria la tarea de realizar una planificación detallando el tiempo estimado necesario para cumplir las distintas actividades y tareas que forman parte del proyecto, así como los recursos asignados a cada una de ellas. Por ello, se ha desarrollado una planificación de trabajo en función del ciclo de vida seleccionado anteriormente, ciclo de vida por prototipos.

Se ha decidido realizar tres iteraciones completas, tras cada una de las cuales el desarrollador se reúne con el tutor para corregir o matizar distintos aspectos de la aplicación.

En lo que se refiere a los recursos humanos que intervienen en el desarrollo del proyecto, a continuación se muestran las personas y roles que participan en el proyecto. Las funciones de las que se encarga cada uno de ellos son:

- **Jefe de Proyecto:** Se encarga de la gestión del proyecto, su organización, planificación y supervisión a lo largo de todo el desarrollo del mismo.
- **Analista de Sistemas:** Se encarga de obtener y redactar los requisitos de usuario, además de modelar los procesos y tareas a codificar.
- **Diseñador:** Su tarea es el diseño de las interfaces y la arquitectura del sistema.
- **Programador:** Se encarga de la codificación del sistema, así como de las pruebas necesarias del mismo.

A continuación se muestra el RBS (Resource Breakdown Structure) del proyecto, una representación de forma jerárquica de los recursos tanto humanos como materiales necesarios para el desarrollo del mismo. Esta técnica de organización de proyectos software tiene por objeto representar la organización humana del proyecto, su estructura, responsabilidades, etc., así como la estructura de recursos tecnológicos y materiales.

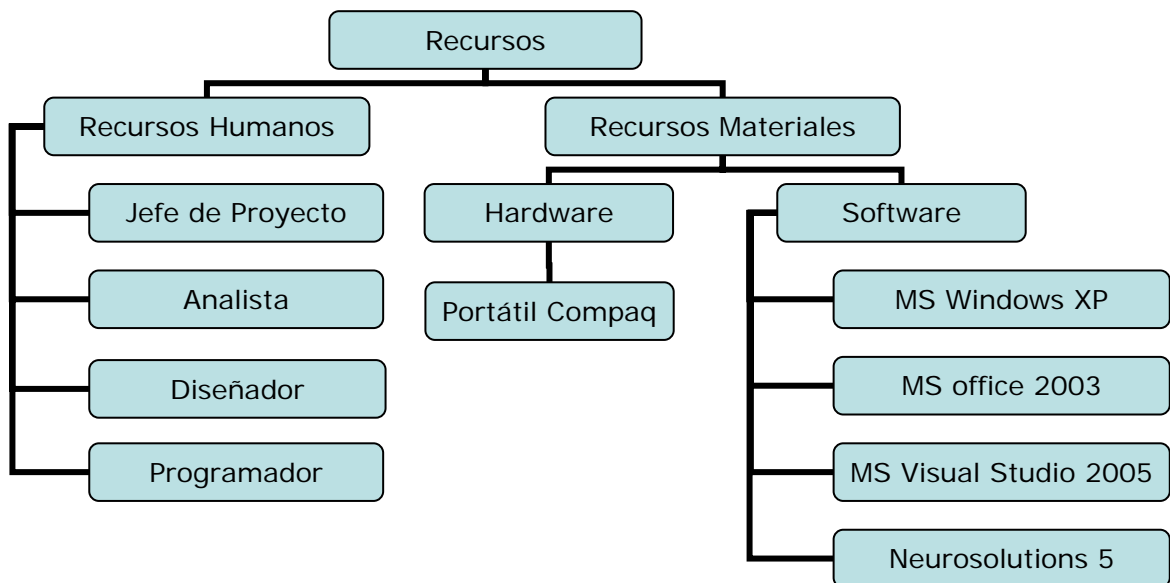


Figura 3. RBS.

El WBS (World Breakdown Structure) es una herramienta simple y práctica para definir el alcance de un proyecto. Es la base para poder saber qué se requiere hacer, para después poder realizar la planificación del proyecto. Se trata de identificar y definir las tareas que se van a realizar en el proyecto.

En el WBS se puede observar la descomposición de los procesos en subprocesos, y éstos a su vez en tareas, agrupadas por funcionalidad.

La base para en la que está basada el WBS es el modelo de proceso software IEEE 1074.

A continuación se muestra la división de las distintas tareas del proyecto mediante el WBS:

El primer proceso que se debe llevar a cabo es el estudio de viabilidad del proyecto, mediante el cual se analiza la solicitud del cliente, la situación actual y si el desarrollo del proyecto es viable. Toda la información se recoge en el documento de Estudio de Viabilidad.

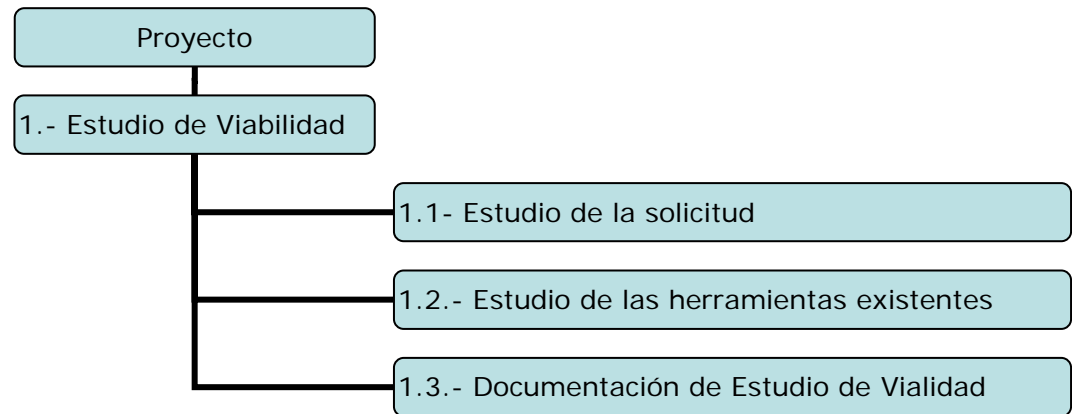


Figura 4. Proceso WBS - Estudio de viabilidad

Una vez que el desarrollo del proyecto se ha considerado viable, el siguiente proceso a realizar es la gestión del proyecto. En este proceso se debe seleccionar el modelo de ciclo de vida más adecuado para el proyecto, y se realiza tanto la planificación a seguir durante el desarrollo del proyecto como una estimación del coste de desarrollo.

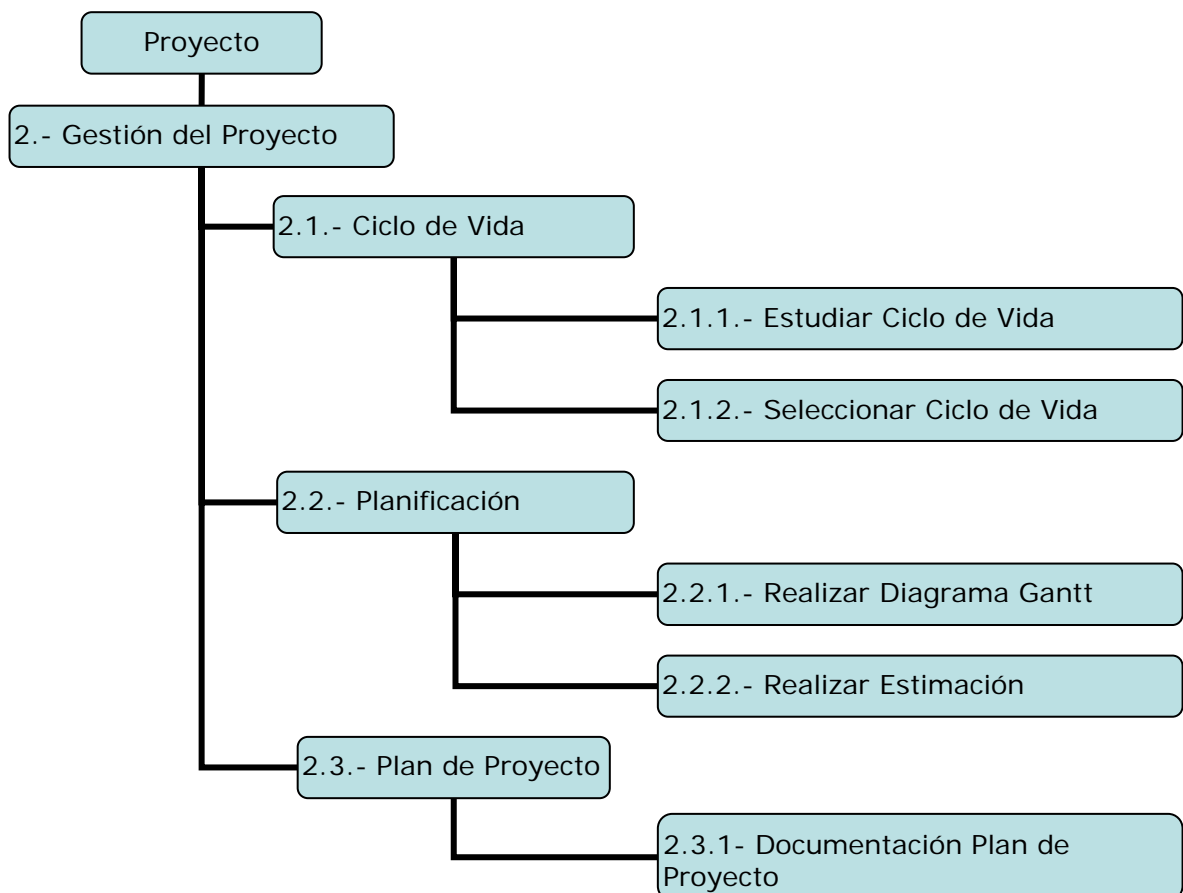


Figura 5. Proceso WBS - Gestión del proyecto

A continuación se presentan las tres iteraciones realizadas en el modelo de ciclo de vida elegido para el desarrollo del sistema. En cada iteración se deben llevar a cabo las tareas propias del análisis, tales como la recopilación de especificaciones y requisitos, o la selección de las herramientas de desarrollo más apropiadas. Es imprescindible la realización de estas tareas antes del diseño, para asegurar que en la fase de diseño se logre un diseño del software eficiente. Además, como parte de la fase de diseño se van a desarrollar las tareas relacionadas con la arquitectura del sistema y la plataforma del proyecto. La fase de implementación consiste principalmente en realizar las tareas de codificación de las interfaces y procesos diseñados previamente. Por último, este proceso consta de un subproceso imprescindible como es el de Pruebas, en el que las tareas a realizar son la definición y ejecución de pruebas para la verificación y validación del software desarrollado.

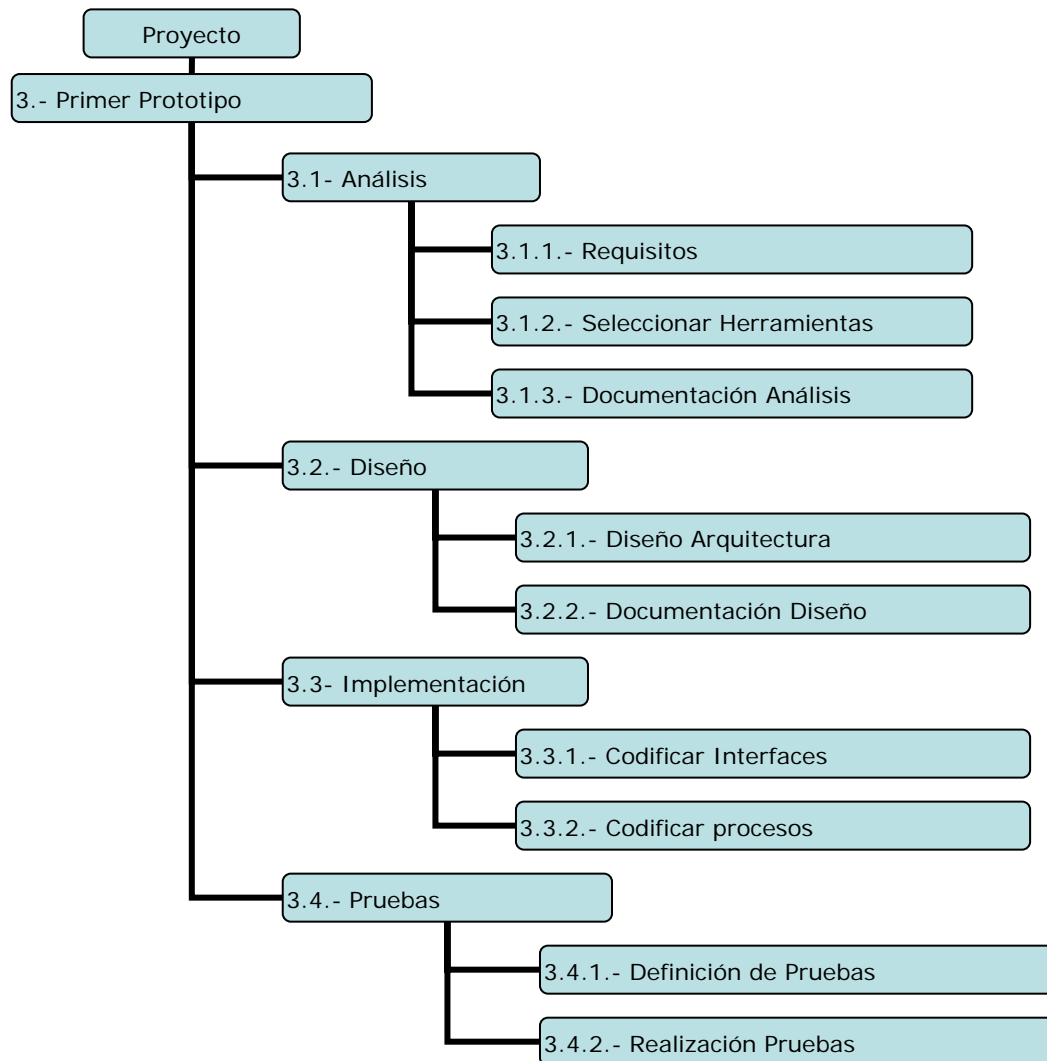


Figura 6. Proceso WBS - Primer Prototipo

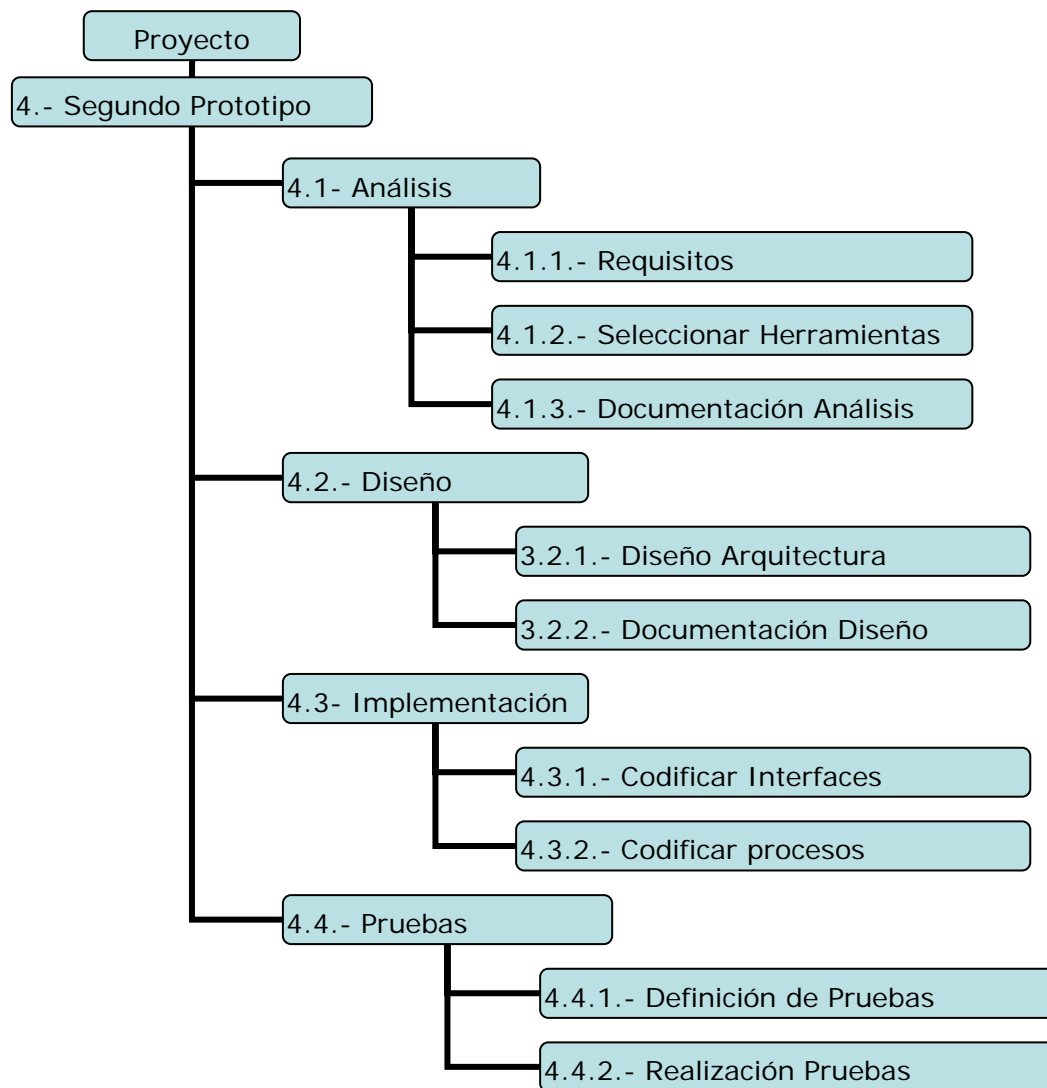


Figura 7. Proceso WBS - Segundo Prototipo

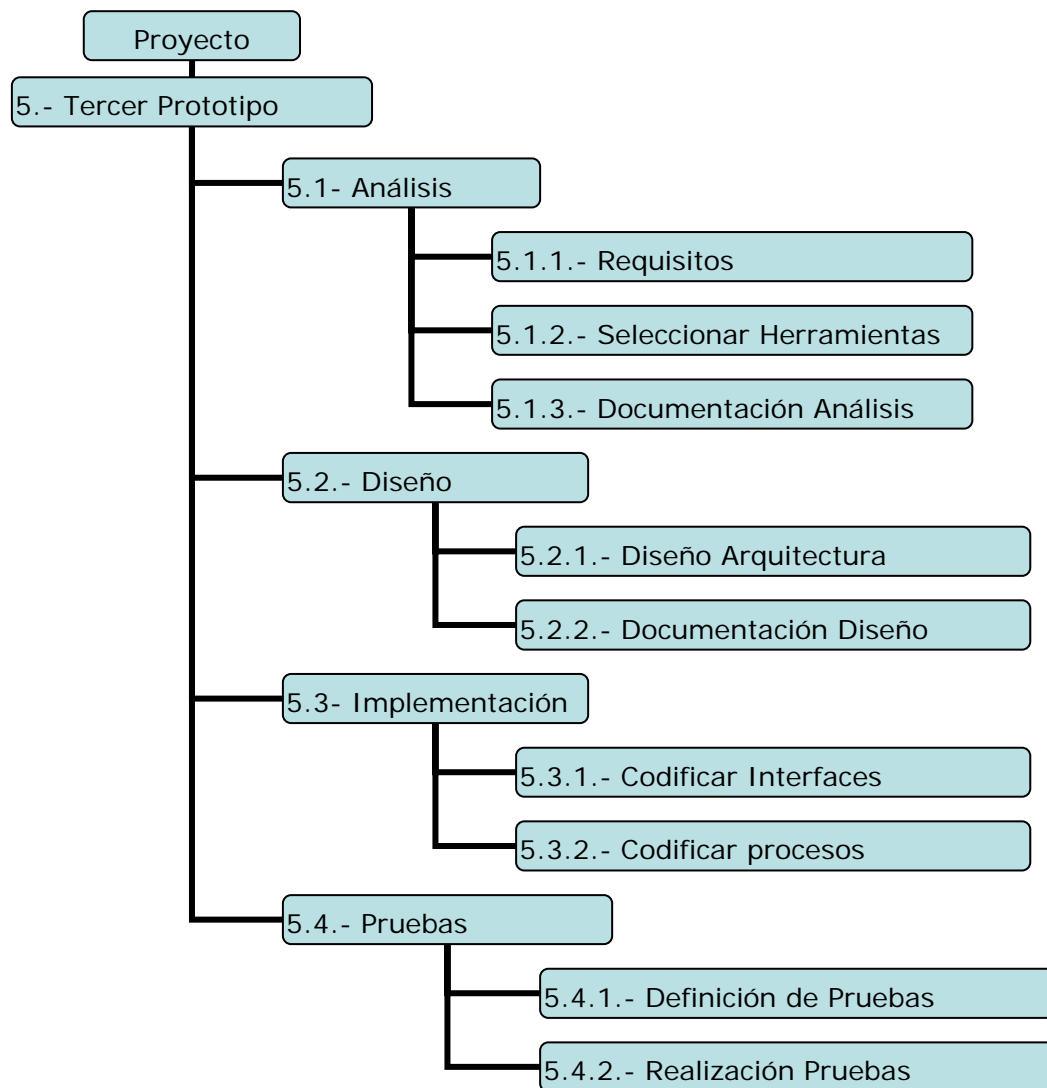


Figura 8. Proceso WBS - Tercer Prototipo

Cuando el último prototipo desarrollado es aprobado por el cliente se pasa al siguiente proceso. En este proceso se realiza el manual de usuario para que éste pueda utilizar todas las funcionalidades disponibles en el sistema con la mayor facilidad.

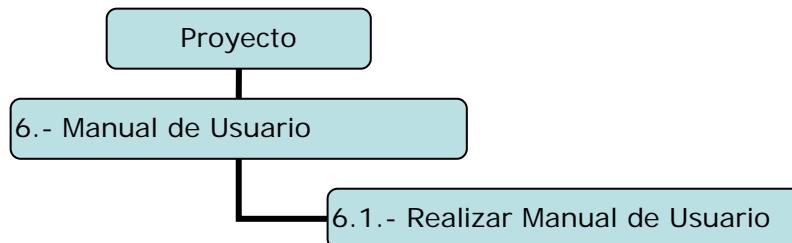


Figura 9. Proceso WBS Manual de Usuario

### 4.3 Planificación

La duración total del proyecto es de 196 días laborables, comenzando su desarrollo el día 4 de Febrero de 2010 y finalizando 15 de Noviembre de 2010, cuando el sistema se encuentre finalizado y documentado para su entrega al cliente.

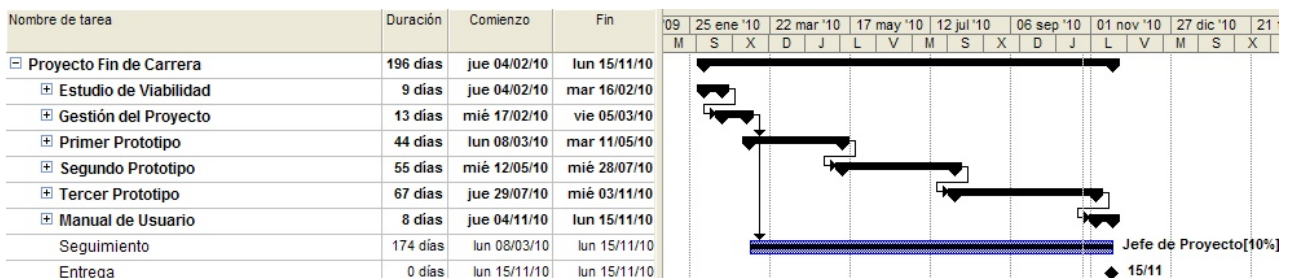


Figura 10. Diagrama de Gantt (General)





## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

El punto de partida del proyecto es el estudio de viabilidad del mismo. Para ello, el jefe de proyecto debe realizar un estudio de la solicitud realizada por el cliente, de la situación actual y elaborar un documento que contenga dicha información. Asimismo, este documento debe indicar si la solicitud realizada es factible de ser realizada o no.

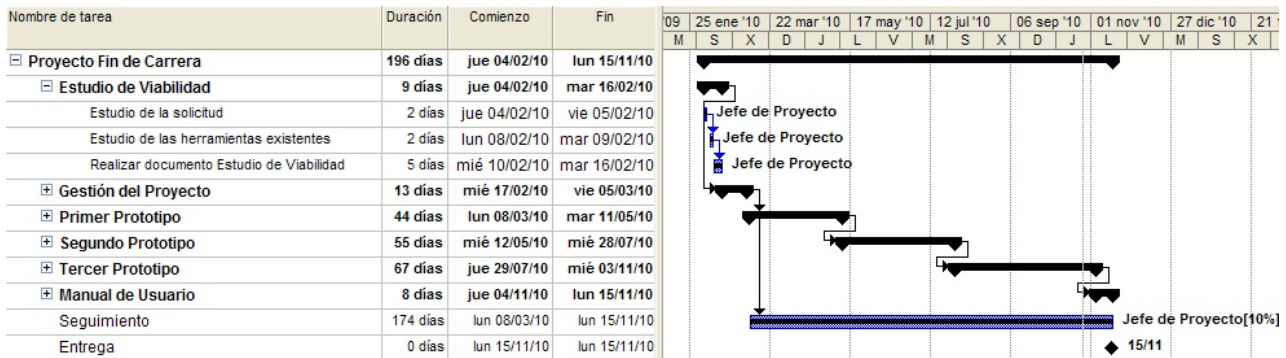


Figura 11. Diagrama de Gantt (Estudio de Viabilidad)

Una vez que el proyecto se ha considerado viable, comienza la fase de gestión del proyecto. De esta fase también se encarga el jefe de proyecto, y en ella debe realizar un estudio sobre los ciclos de vida existentes, seleccionar el más apropiado para el desarrollo del presente proyecto, documentando claramente las razones y motivos de su elección.

A continuación, se realiza el diagrama Gantt en el cual figurarán todas las tareas y actividades necesarias para realizar el desarrollo de la aplicación, así como la duración estimada de cada una de ellas y los recursos necesarios que se utilizaran en cada una de las tareas. En este diagrama las tareas ya aparecen como finalizadas, puesto que ya se ha concluido el proyecto.

Por último, el jefe de proyecto ha de realizar el correspondiente documento de Gestión de Proyecto, en el que se incluyan toda la planificación y estimaciones realizadas y comienza el seguimiento de éste, el seguimiento se llevará a cabo en paralelo a todas las actividades a realizar hasta la entrega final de la aplicación.



En la siguiente figura se muestran las tareas que conforman la Gestión de Proyecto:

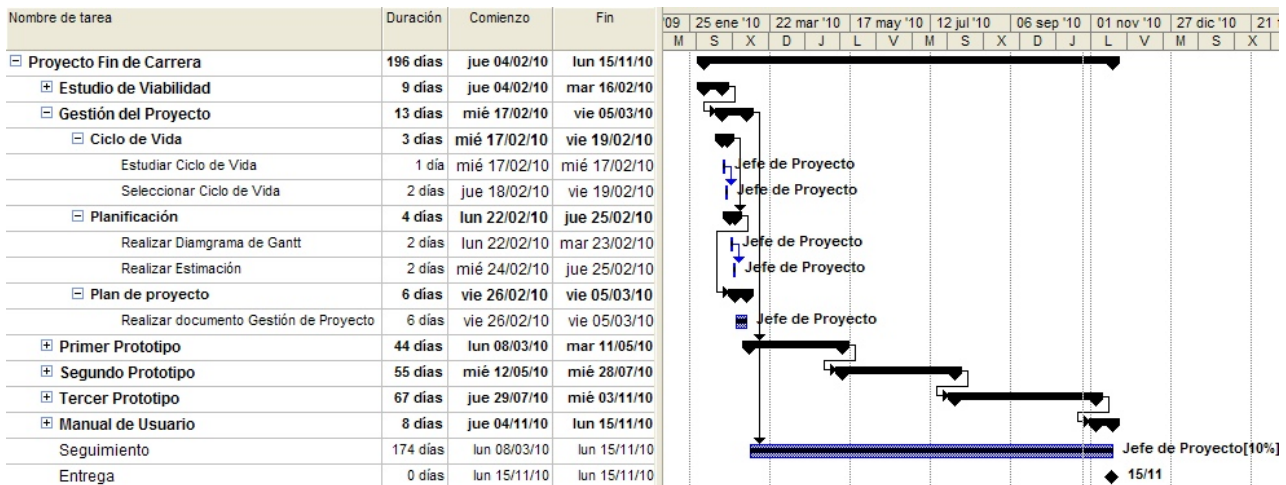


Figura 12. Diagrama de Gantt (Gestión del Proyecto)

Cuando todas las tareas de gestión del proyecto han sido realizadas comienza el desarrollo del primer prototipo del sistema. En primer lugar el analista debe recopilar los requisitos del usuario para obtener los requisitos software y seleccionar las herramientas necesarias para satisfacerlos. Es necesario que antes de crear un prototipo, el analista trabaje en conjunto con el cliente o tutor para lograr una correcta identificación de los requerimientos a satisfacer.

El diseñador lee la documentación generada por el analista y diseña la arquitectura del sistema, definiendo para ello tanto las interfaces como los procesos a implementar.

A continuación el programador debe implementar el sistema de acuerdo al diseño realizado para satisfacer los requisitos del usuario lo más fielmente posible al diseño realizado.

El primer prototipo se concluye con una batería de pruebas, para verificar su correcto funcionamiento y la evaluación por parte del cliente o tutor. Es responsabilidad del cliente trabajar con el prototipo y evaluar sus características y operación. La experiencia del sistema, bajo condiciones reales, permite obtener la familiaridad indispensable para determinar los cambios o mejoras que sean necesarios, así como las características inadecuadas.



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Las pruebas serán redactadas por el diseñador y ejecutadas por el programador para comprobar el correcto funcionamiento de la aplicación.

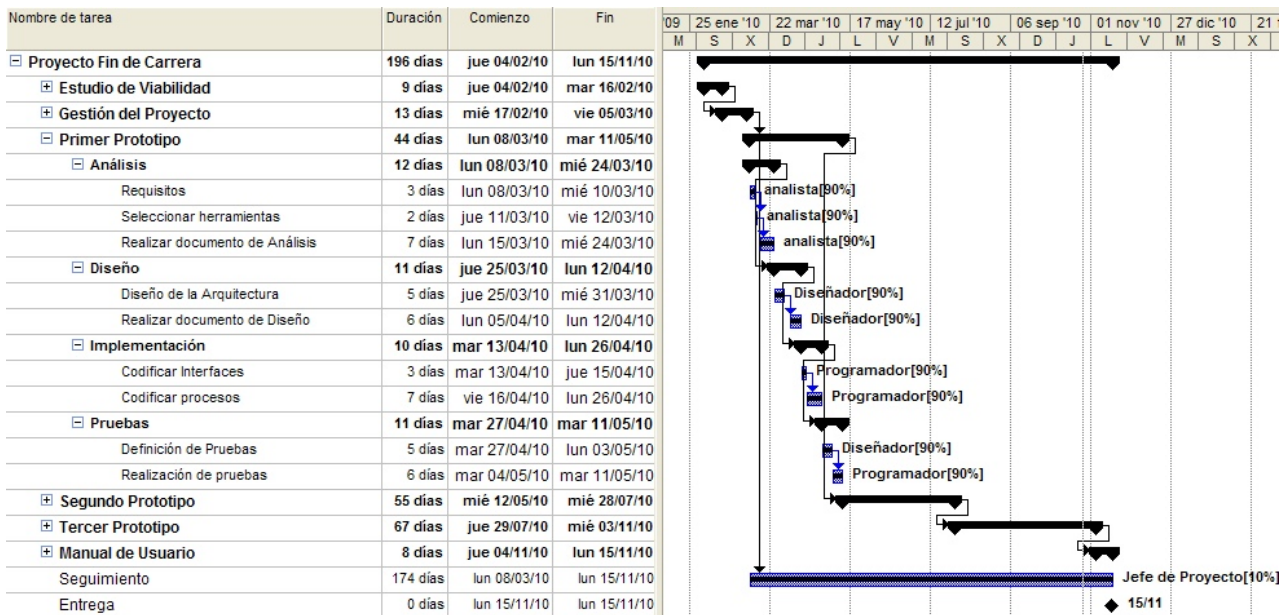


Figura 13. Diagrama de Gantt (Primer Prototipo)

El analista comienza la segunda iteración partiendo de la información aportada por el usuario en la evaluación del primer prototipo. Los cambios al prototipo son planificados con el cliente antes de llevarlos a cabo, aunque es el analista el responsable de tales modificaciones.

Como en el primer prototipo, el analista documenta los nuevos requisitos y selecciona las herramientas necesarias. A continuación el diseñador lee la documentación aportada por el analista y diseña las nuevas interfaces y funcionalidades, las cuales el programador codifica de acuerdo a las especificaciones.

Se realizan las pruebas para verificar que todo funciona correctamente y se le muestra al cliente para que evalúe el nuevo prototipo.



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

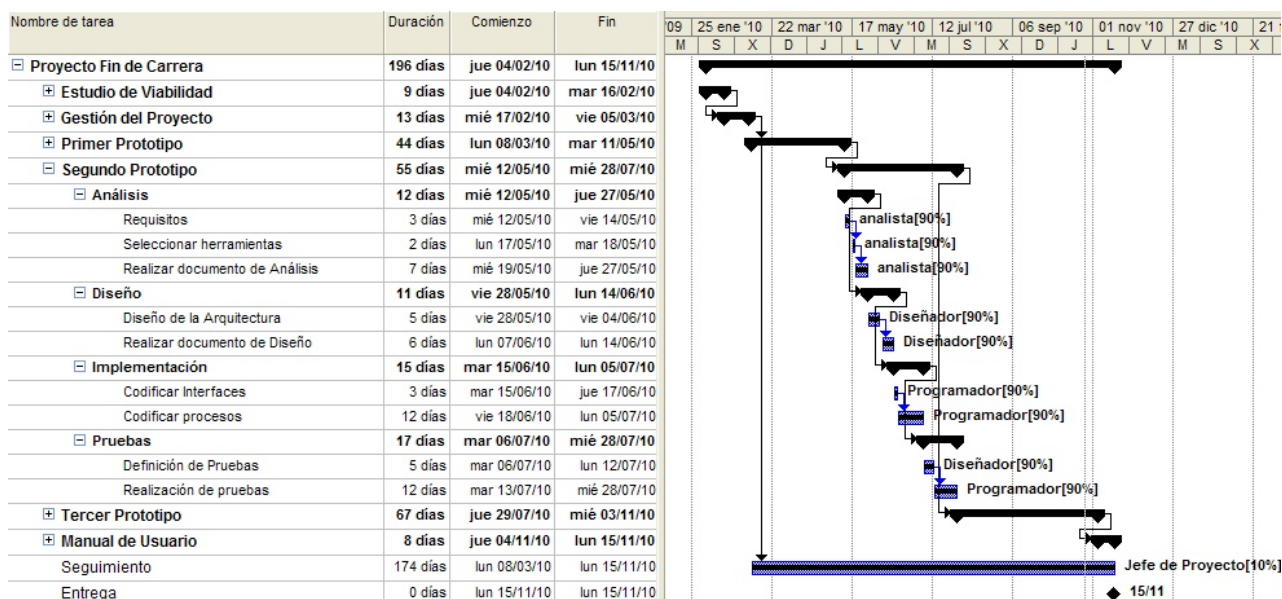


Figura 14. Diagrama de Gantt (Segundo Prototipo)

Finalmente se comienza con el tercer y último prototipo, en el que se intentarán satisfacer todos los requisitos expuestos por el cliente a lo largo del proyecto. El analista recoge todos los requisitos y comentarios del usuario y elabora la documentación definitiva, si es necesario utilizar una nueva herramienta de desarrollo la buscará y seleccionará. El diseñador plantea las interfaces y funcionalidades que sean necesarias y el programador implementa la aplicación definitiva.

La fase de pruebas es crucial para determinar que los requerimientos han sido satisfechos y que el sistema se comporta como se esperaba. El prototipo debe pasar las pruebas de funcionalidad para validar su correcto funcionamiento y la evaluación por parte del cliente, siendo éste el que certifique la aceptación del sistema que utiliza.





## Proyecto Fin de Carrera

Miguel Fortes Gómez  
Ingeniería Técnica en Informática de gestión

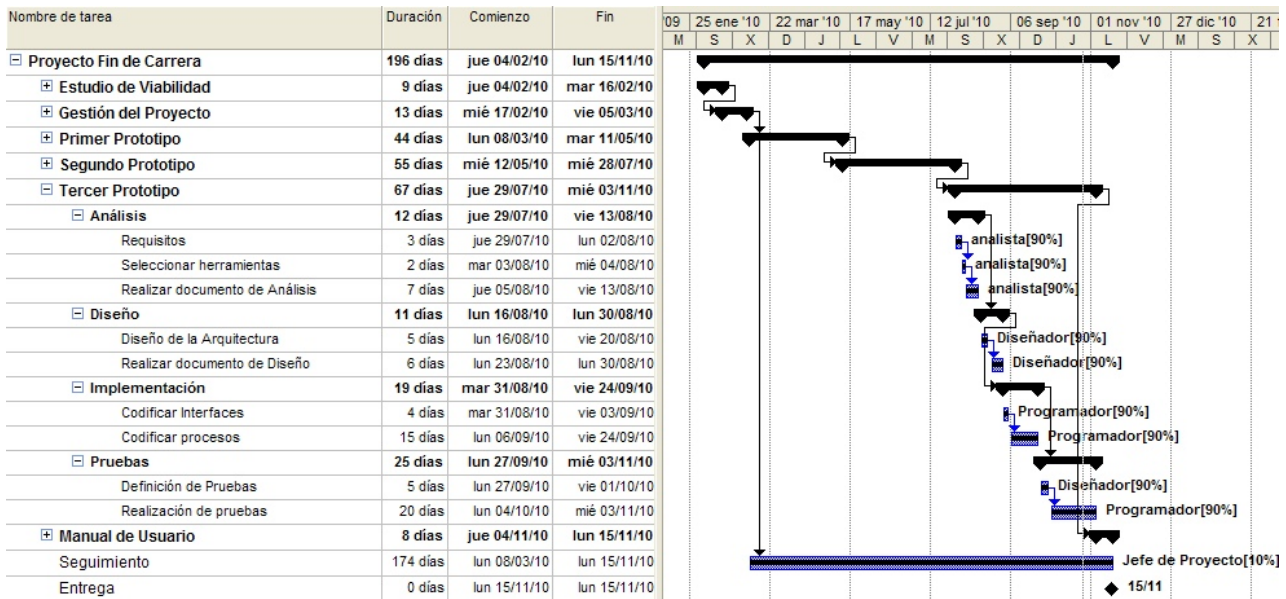


Figura 15. Diagrama de Gantt (Tercer Prototipo)

Finalmente se genera un manual de usuario indicando todos los aspectos de la aplicación, tanto sus funciones como la manera de ejecutarlas o seleccionarlasy, para que el usuario final u otra persona sea capaz de utilizar la nueva aplicación sin problemas, y no tener que recurrir a los desarrolladores.

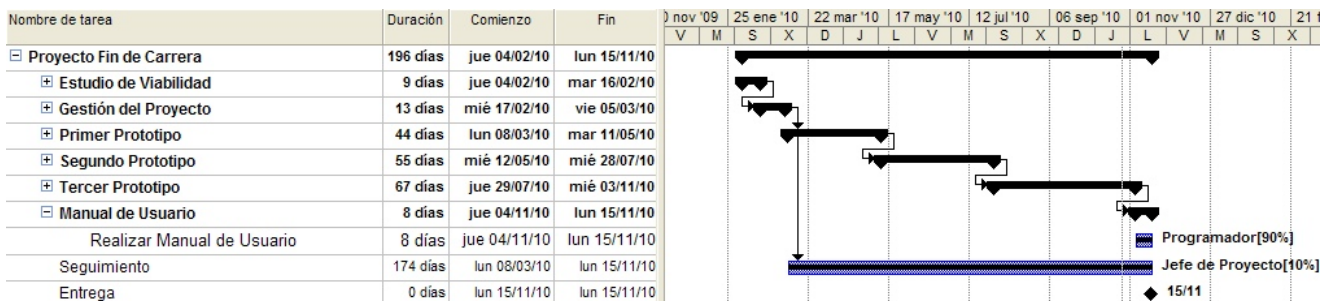


Figura 16. Diagrama de Gantt (Manual de Usuario)



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

El último hito del proyecto es la entrega de la documentación y el software desarrollado al cliente. Tras este paso se da por concluido el proyecto el día 15 de Noviembre de 2010.

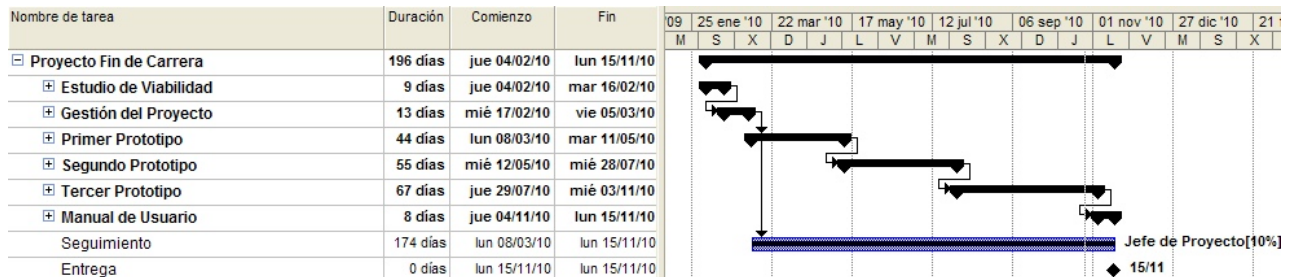


Figura 17. Diagrama de Gantt (Entrega)



#### **4.4 Estimación de costes**

Para calcular el coste estimado del proyecto se han tenido en cuenta tanto los costes materiales como los costes humanos del mismo.

En primer lugar se calcula el coste asociado a las herramientas utilizadas para el desarrollo del proyecto:

Herramienta	Coste
Microsoft Office 2003	379 €
Microsoft Windows XP	199 €
Microsoft Visual Studio 2005	550 €
Neurosolutions 5.0	573 €
<b>Total (IVA Incluido)</b>	<b>1.701 €</b>

Tabla 41. Coste Licencias

En cuanto al coste de recursos materiales se ha contabilizado, además, la cantidad de 800€ en cuanto a equipamiento informático.

Para el cálculo de los costes asociados a los recursos humanos se ha tomado una jornada laboral de 8 horas, distribuidas en horario de 9:00 a 14:00 y 16:00 a 19:00. El proyecto ha sido desarrollado por un solo ingeniero, el cual ha adoptado los distintos roles anteriormente descritos. La siguiente tabla muestra una relación entre las funciones que se han adoptado, el tiempo invertido en cada una de ellas y su coste:

Recurso	Horas	Precio/Hora	Total
Jefe de proyecto	315,2	45 €	14.184 €
Analista	259,2	35 €	9.072€
Diseñador	345,6	35 €	12.096 €
Programador	648	25 €	16.200€
<b>Total</b>	<b>1568</b>		<b>51.552 €</b>

Tabla 42. Coste recursos humanos

Analizando la tabla anterior se puede observar que el coste total resultante de las horas dedicadas por el desarrollador asciende a 51.552 €.



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Por lo tanto, el coste total de desarrollo del proyecto es la suma de los costes materiales, más los costes humanos:

Coste Recursos	Coste
Coste de Recursos materiales	2.501 €
Coste de recursos Humanos	51.552 €
<b>Total (IVA Incluido)</b>	<b>54.053 €</b>

Tabla 43. Coste total del proyecto

El coste total de desarrollo del proyecto asciende a la cantidad de 54.053 € (cincuenta y cuatro mil cincuenta y tres euros).





## **5 Análisis del Sistema**

### **5.1 Introducción**

#### **5.1.1 Objetivo del Análisis del Sistema**

Al realizar el Análisis del Sistema se pretende obtener una colección completa y detallada de los requisitos del sistema, tomando como punto de partida los requisitos identificados en el Estudio de Viabilidad del Sistema.

El documento generado en esta fase será la base para el desarrollo de la fase de Diseño del Sistema, en el que se especificará el diseño completo del sistema de información.

#### **5.1.2 Alcance**

Esta fase tiene como objetivo obtener una especificación detallada del sistema que se va a diseñar. Mediante su producto, el Documento de Análisis del Sistema, se pretende captar cuales son las necesidades que tiene el cliente.

En primer lugar, se definirá cual es el alcance del sistema que se desea desarrollar, así como el entorno tecnológico asociado al proyecto. Además, se identificarán los diferentes participantes que aparecen a lo largo de la vida del proyecto así como los usuarios finales.

A continuación, se definirán cuales son los requisitos software que debe cumplir el sistema a desarrollar, tomando como punto de partida los casos de uso y los requisitos de usuario especificados en la anterior fase, el Estudio de Viabilidad del Sistema.

Por último, se definirán las interfaces de usuario que se utilizarán, dejando como última tarea las comprobaciones de calidad sobre los distintos modelos y requisitos software que se han generado durante la fase de análisis.



## **5.2 Definición del Sistema**

### **5.2.1 Determinación del alcance del sistema**

En este apartado se determina el alcance del sistema a desarrollar, para que permita satisfacer las necesidades planteadas por el cliente y recogidas en el documento de Estudio de la Viabilidad.

El sistema a desarrollar consiste en una aplicación que permite la automatización del proceso de aprendizaje de redes neuronales artificiales

La siguiente aplicación debe cumplir las siguientes características:

- Interaccionar con una aplicación de redes de neuronas artificiales.
- Generar informes de evaluación, para tratamiento en local
- Controlar errores en la introducción de parámetros de entrada por parte del usuario.
- Entrenamiento de experimentos en varias tipologías de redes neuronales.
- Permitir la entrada de datos multidominio.

Asimismo, de forma complementaria la aplicación también debe ser capaz de tratar los ficheros de entrada, preparando todos los ficheros de datos para los experimentos que se deseen realizar con el sistema.

### **5.2.2 Identificación del entorno tecnológico**

Véase el apartado 5.2.3.3. Entorno operacional.



### 5.2.3 Especificación de estándares y normas

Los estándares y normas que ha de respetar el proceso de desarrollo del presente proyecto son los siguientes:

- **Métrica Versión 3:** Los documentos Estudio de Viabilidad del Sistema, Análisis del Sistema y Diseño del Sistema generados durante el proyecto seguirán una adaptación de la metodología de desarrollo Métrica Versión 3 [8]. La metodología será adaptada a las necesidades del proyecto, de manera que determinadas tareas no se realizarán o no se profundizará demasiado en ellas.
- **UML:** Es el lenguaje de modelado de sistemas que se utilizará para realizar el diseño del sistema.
- **IEEE 1074:** Es la norma utilizada en la Gestión del Proyecto para la definición del modelo de procesos planificado para el desarrollo del proyecto.



### **5.2.3.1 Restricciones generales**

A continuación se detallan las restricciones que deberá cumplir el sistema a diseñar:

En primer lugar, el sistema funcionará bajo sistemas operativos Microsoft Windows XP únicamente, por lo que la máquina para la ejecución del sistema deberá cumplir esta restricción.

La interfaz, en los módulos del sistema que la incluyan, con la que interaccionará el usuario final, debe proporcionar todas las funcionalidades descritas en los requisitos software del sistema, de una forma amigable e intuitiva, evitando posibles ambigüedades que puedan ocasionar confusión al usuario.

El idioma usado en el desarrollo del sistema será el castellano. Los usuarios tienen libertad de expresarse en el idioma que consideren oportuno, pero la interfaz de la aplicación estará en castellano.

### **5.2.3.2 Supuestos y dependencias**

Todos los usuarios participantes en la presente fase de análisis conocen el estándar empleado Métrica Versión 3. Por este motivo, todos los productos generados durante el proceso deben respetar lo establecido en dicho estándar.

### **5.2.3.3 Entorno operacional**

El sistema se desarrollará mediante el entorno de desarrollo Microsoft Visual Studio 2005, utilizando como lenguaje de programación Visual Basic .NET. Los usuarios del sistema únicamente tendrán que disponer de un ordenador con sistema operativo Microsoft Windows XP.

Para el desarrollo del proyecto se ha utilizado un ordenador con las siguientes características:

- **Ordenador portátil Compaq:**

Microprocesador: AMD Athlon™ 64 Processor 3200+ 1.99 GHz.

Memoria RAM: 1024 mb.

Sistema Operativo: Microsoft Windows XP.



Suite ofimática Microsoft Office 2003 para la gestión de documentos.

### **5.2.3.4 Identificación de los usuarios**

En este apartado se van a detallar los usuarios que participan en el proceso de análisis del sistema, así como los usuarios que lo validarán y aceptarán finalmente.

En primer lugar, en el proceso de análisis van a participar:

- **Equipo de desarrollo:** Grupo de personas encargado de llevar a cabo el desarrollo del presente proyecto.
- **Cliente o tutor:** El cliente o en este caso tutor del proyecto, que esperan un correcto desarrollo del producto final en términos de calidad, tiempo y coste. Además, son parte importante a la hora de la obtención de los requisitos del sistema a diseñar, con el objetivo de obtener un producto que cumpla con las necesidades que le han llevado a solicitar el sistema.

En cuanto a los usuarios finales del sistema, éstos serán los que se describen a continuación:

- **Usuarios gestores:** Son aquellas personas que harán uso del sistema desarrollado. Se encargarán tanto de la configuración del sistema como de su puesta en producción.

### **5.2.3.5 Estudio de la seguridad requerida en el proceso de análisis**

El acceso a los productos generados en la presente fase de análisis estará limitado a los usuarios participantes que se describen en el apartado 5.2.3.4. Identificación de los usuarios.



## 5.3 *Establecimiento de Requisitos Software*

### 5.3.1 Obtención de requisitos

A continuación se presentan los requisitos de software identificados en el análisis. El origen de estos requisitos se encuentra en los requisitos de usuario definidos en el Estudio de Viabilidad del Sistema, junto con decisiones tomadas por el equipo de desarrollo.

Los tipos de Requisitos de Software identificados en el proceso han sido los siguientes:

- **Requisitos funcionales (F):** Son aquellos requisitos que especifican el propósito del sistema, los cuales derivan directamente de los casos de uso.
- **Requisitos de interfaz (I):** Especifican hardware y/o software con los que debe interactuar el sistema.
- **Requisitos de operación (O):** Son todos aquellos requisitos que especifican como se va a resolver el problema.
- **Requisitos de documentación (D):** Requisitos que especifican criterios de realización de la documentación asociada al proyecto.
- **Requisitos de calidad (C):** Estos requisitos especifican los atributos del software que aseguran que serán adecuados para su propósito.

Cada requisito software debe incluir en su definición, además de la descripción del mismo, una serie de atributos, los cuales proporcionen toda la información necesaria para su seguimiento posterior y su clasificación. Estos atributos se describen a continuación:

- **Identificador:** Cada requisito debe estar identificado de forma única. Este identificador tendrá el siguiente formato: *RS-XX-nnn* o *RS-XX-nnn*, donde:
  - RS: Indica que se trata de un requisito software.
  - XX: Indicará el tipo de requisitos software al que corresponde (ver listado anterior).



- nnn: Tomará valores numéricos dentro del rango 000-999.
- **Prioridad:** Se asignará una prioridad a cada requisito con el fin de poder realizar una planificación correcta durante fases posteriores. Su clasificación puede tomar los valores: alta, media y baja.
- **Necesidad:** Los requisitos clasificados como esenciales no pueden ser eliminados, mientras que los demás requisitos estarán siempre sujetos a modificación en el caso de que exista una causa que lo justifique. Por lo tanto, se clasificarán descendientemente de acuerdo a su necesidad de la siguiente forma: esencial, deseable y opcional.
- **Claridad:** Identifica la falta o existencia de ambigüedad de un requisito, esto es, si puede ser interpretado de varias formas dependiendo del contexto. Los requisitos serán clasificados de acuerdo a su claridad de forma descendente de la siguiente forma: alta, media y baja.
- **Fuente:** Identifica el origen del requisito, que puede estar en un requisito de usuario, o en el propio equipo de desarrollo del proyecto.
- **Estabilidad:** Algunos requisitos pueden no estar sujetos a cambios durante el proyecto debido a su naturaleza, mientras que otros pueden estar sujetos a determinados cambios por el desarrollo de la etapa de diseño. Los requisitos poco estables deberán ser observados con mayor cuidado durante el desarrollo del proyecto al ser susceptibles a cambios. Cada requisito se clasificará en: estable o inestable.
- **Verificabilidad:** Indica si el cumplimiento de un requisito en el sistema puede ser susceptible de comprobación, esto es, si se puede verificar que el requisito se ha incorporado en el diseño y que en el sistema se puede verificar su cumplimiento. La verificabilidad de cada requisito se clasificará en: alta, media y baja.



### 5.3.1.1 Requisitos funcionales

Identificador	RS-F-001		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-001
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Iniciar la aplicación de forma que el usuario no deba borrar información introducida previamente, esto es, con todos sus elementos de información en su estado inicial.		

Tabla 44. Requisito software RS-F-001

Identificador	RS-F-002		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-002
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Seleccionar una ruta del fichero de entrada mediante un cuadro de diálogo de selección de archivos.		

Tabla 45. Requisito software RS-F-002





Identificador	RS-F-003		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-002
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Seleccionar una ruta de salida mediante dos posibilidades: <ul style="list-style-type: none"><li>• Cuadro de diálogo de selección de carpetas.</li><li>• Introducción manual de la ruta completa.</li></ul>		

Tabla 46. Requisito software RS-F-003

Identificador	RS-F-004		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-003
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Importar datos desde un fichero en formato Microsoft Excel, pudiendo seleccionar la hoja en la que se encuentran los datos, y los atributos que se desean importar.		

Tabla 47. Requisito software RS-F-004



Identificador	RS-F-005		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-004
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Mostrar las instancias disponibles en el fichero de entrada y las instancias pendientes de repartir.		

Tabla 48. Requisito software RS-F-005

Identificador	RS-F-006		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-004
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Permitir el reparto de instancias asignando porcentajes a cada una de las clases existentes para un "Atributo de reparto" seleccionado entre los "Atributos del fichero de datos". Dichos porcentajes los puede asignar el usuario de forma manual o bien asignarse de forma aleatoria.		

Tabla 49. Requisito software RS-F-006



Identificador	RS-F-007		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-004
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Repartir las instancias disponibles en tres conjuntos: entrenamiento, test y validación, siendo opcional la inclusión de este último en los ficheros generados. Se deberá tener en cuenta la existencia o no de un "Atributo de reparto", y de los porcentajes de reparto asignados a cada clase, en caso de existir.		

Tabla 50. Requisito software RS-F-007

Identificador	RS-F-008		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-004
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Permitir la opción de "Forzar reparto", para que la aplicación alcance las cantidades indicadas para los conjuntos de entrenamiento, test y validación, a pesar de no cumplir los porcentajes de reparto de clases establecidos.		

Tabla 51. Requisito software RS-F-008



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RS-F-009		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-005
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Disponer de las siguientes técnicas de división de instancias: <ul style="list-style-type: none"><li>Validación cruzada k-fold (KFCV).</li><li>Validación cruzada leave-one-out (LOOCV).</li><li>Remuestreo bootstrap (RBP).</li><li>Variante de KFCV (VKFCV).</li></ul>		

Tabla 52. Requisito software RS-F-009

Identificador	RS-F-010		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-006
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Añadir experimentos a la lista de "Experimentos a preparar" con los parámetros de configuración seleccionados. Dichos parámetros de configuración son: <ul style="list-style-type: none"><li>Técnica de división.</li><li>Número de pruebas.</li><li>Atributo de reparto.</li><li>Atributos de salida.</li><li>Porcentajes de reparto de instancias según las clases existentes.</li></ul>		

Tabla 53. Requisito software RS-F-010



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RS-F-011		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-007
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	<p>Añadir experimentos a la lista de "Experimentos a preparar" con los parámetros de configuración seleccionados, incrementando o decrementando de forma automática los valores deseados por el usuario. Dichos parámetros de configuración son:</p> <ul style="list-style-type: none"><li>• Técnica de división.</li><li>• Número de pruebas.</li><li>• Atributo de reparto.</li><li>• Atributos de salida.</li><li>• Porcentajes de reparto de instancias según las clases existentes.</li></ul>		

Tabla 54. Requisito software RS-F-011

Identificador	RS-F-012		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-008
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	<p>Eliminar experimentos añadidos previamente a la lista de "Experimentos a preparar", pudiendo eliminarse múltiples experimentos de una vez.</p>		

Tabla 55. Requisito software RS-F-012



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RS-F-013		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-009
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	<p>Guardar la configuración de la aplicación en un momento dado, almacenando las siguientes características:</p> <ul style="list-style-type: none"><li>• Ruta del fichero de entrada.</li><li>• Ruta de salida.</li><li>• Lista de "Experimentos a preparar".</li><li>• Lista de "Atributos del fichero de datos".</li><li>• Porcentajes de reparto de las clases, o marcador de reparto aleatorio en su defecto.</li><li>• Marcador de "Forzar reparto".</li></ul>		

Tabla 56. Requisito software RS-F-013

Identificador	RS-F-014		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-009
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	<p>Cargar la configuración de la aplicación almacenada previamente, recuperando las siguientes características:</p> <ul style="list-style-type: none"><li>• Ruta del fichero de entrada.</li><li>• Ruta de salida.</li><li>• Lista de "Experimentos a preparar".</li><li>• Lista de "Atributos del fichero de datos".</li><li>• Porcentajes de reparto de las clases, o marcador de reparto aleatorio en su defecto.</li><li>• Marcador de "Forzar reparto".</li></ul>		

Tabla 57. Requisito software RS-F-014



Identificador	RS-F-015		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-010
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Generar los archivos de entrenamiento, validación y test correspondientes a los experimentos situados en la lista "Experimentos a preparar". La denominación de estos archivos seguirá el formato: "TrainXX.txt", "ValXX.txt", y "TestXX.txt", donde XX representa un valor numérico secuencial.		

Tabla 58. Requisito software RS-F-015

Identificador	RS-F-016		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-011
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Cerrar la aplicación mediante el botón "Salir" situado en la interfaz.		

Tabla 59. Requisito software RS-F-016



Identificador	RS-F-017		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-012
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Seleccionar una ruta de localización de los ficheros de red mediante dos posibilidades: <ul style="list-style-type: none"><li>• Cuadro de diálogo de selección de carpetas.</li><li>• Introducción manual de la ruta completa.</li></ul>		

Tabla 60. Requisito software RS-F-017

Identificador	RS-F-018		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-013
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Mostrar los ficheros de red y la tipología de red identificada por la aplicación pudiendo cambiar la tipología de la red de un fichero de red.		

Tabla 61. Requisito software RS-F-018





Identificador	RS-F-019		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-014
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Seleccionar el criterio de parada para el entrenamiento. Pudiendo modificar el numero de ciclos (epochs) máximos de la simulación, el MSE o si los experimentos son con validación cruzada tenerla en cuenta para el MSE.		

Tabla 62. Requisito software RS-F-019

Identificador	RS-F-020		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-015
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	<p>Interaccionar con NeuroSolutions para realizar las siguientes funciones:</p> <ul style="list-style-type: none"><li>• Entrenar varias redes ya creadas.</li><li>• Validar varias redes entrenadas.</li><li>• Poner varias redes en producción.</li></ul> <p>En cualquier caso la predicción a realizar podrá ser tanto Clasificación como Regresión.</p>		

Tabla 63. Requisito software RU-F-020



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	RS-F-021		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-016
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Generar informes de evaluación en formato de Microsoft Excel, incluyendo: <ul style="list-style-type: none"><li>• Clasificación:<ul style="list-style-type: none"><li>◦ Rangos de valores de entrenamiento.</li><li>◦ Tasa de aciertos y fallos.</li><li>◦ Influencia de las variables.</li><li>◦ Variables críticas.</li></ul></li><li>• Regresión:<ul style="list-style-type: none"><li>◦ Coeficientes de correlación de las variables de salida.</li></ul></li></ul>		

Tabla 64. Requisito software RU-F-021

Identificador	RS-F-022		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-C-017
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Generar informes de resumen en formato de Microsoft Excel, incluyendo: <ul style="list-style-type: none"><li>• Clasificación:<ul style="list-style-type: none"><li>◦ Tasa total de aciertos y fallos</li><li>◦ Influencia Total de las Variables</li></ul></li><li>• Regresión:<ul style="list-style-type: none"><li>◦ Coeficientes de correlación totales de las variables de salida.</li></ul></li></ul>		

Tabla 65. Requisito software RU-F-022



### 5.3.1.2 Requisitos de interfaz

Identificador	RS-I-001		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-R-001
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Generar archivos en formato de texto plano y con extensión ".txt".		

Tabla 66. Requisito software RS-I-001

### 5.3.1.3 Requisitos de operación

Identificador	RS-O-001		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-R-002
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Disponer de una interfaz sencilla e intuitiva que permita configurar todos los parámetros de experimentación de forma rápida para el usuario.		

Tabla 67. Requisito software RS-O-001



Identificador	RS-O-002		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-R-003
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Mostrar el progreso de la generación de archivos mediante una barra de progreso situada en la interfaz de usuario de la aplicación.		

Tabla 68. Requisito software RS-O-002

Identificador	RS-O-003		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-R-004
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Almacenar los archivos de configuración de la aplicación en formato de Microsoft Excel. Este proceso debe ser transparente al usuario, es decir, el usuario simplemente pulsará un botón "Guardar configuración" y la aplicación se encargará de todo el proceso.		

Tabla 69. Requisito software RS-O-003



Identificador	RS-O-004		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-R-005
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Desarrollar la aplicación en lenguaje Visual Basic.NET mediante el entorno de desarrollo Microsoft Visual Studio 2005.		

Tabla 70. Requisito software RS-O-004

Identificador	RS-O-005		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	RU-R-006
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Disponer de una interfaz en la que se muestre la información del entrenamiento de los experimentos con las redes seleccionadas		

Tabla 71. Requisito software RS-O-005



### 5.3.1.4 Requisitos de documentación

Identificador	RS-D-001		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Equipo de desarrollo
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Proporcionar al cliente un manual de usuario impreso, en el que se incluya una descripción detallada de las principales funcionalidades del sistema. El idioma del manual es castellano.		

Tabla 72. Requisito software RS-D-001

### 5.3.1.5 Requisitos de calidad

Identificador	RS-C-001		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Equipo de desarrollo
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Comprobar la adecuación de todos los documentos entregados a lo largo del proyecto al estándar establecido para este desarrollo software.		

Tabla 73. Requisito software RS-C-001



Identificador	RS-C-002		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Equipo de desarrollo
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Realizar copias de seguridad o backups semanales. Estas copias semanales serán totales, es decir, se realizarán sobre todo el trabajo realizado en esa semana.		

Tabla 74. Requisito software RS-C-002



## **5.4 Identificación de subsistemas de análisis**

No existen subsistemas ya que la aplicación a construir se considera como un único bloque que realiza toda la funcionalidad.





## **5.5 *Análisis de los casos de uso***

En este apartado se realiza un estudio de los casos de uso identificados en el Estudio de Viabilidad del Sistema, con el fin de identificar las clases necesarias para el sistema, así como las relaciones entre ellas y su asociación con los casos de uso.

El objetivo es identificar el mínimo número de clases que serán necesarias para el sistema, pero suficientes para cubrir todos los requisitos especificados.



### 5.5.1 Identificación de clases asociadas a un caso de uso

En los siguientes apartados se relacionan los casos de uso de cada uno de los módulos del sistema con las clases asociadas a cada uno de ellos.

Identificador	Clases Asociadas
CU-001	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-002	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• Interfaz importación.</li></ul>
CU-003	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-004	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-005	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-006	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• GestorExperimento.</li><li>• DivisionKFCV.</li><li>• DivisionLOOCV.</li><li>• DivisionRBP.</li><li>• DivisionVKFCV.</li></ul>
CU-007	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• Gestor Configuración.</li></ul>
CU-008	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• Gestor Configuración.</li></ul>
CU-009	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-010	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• InterfazEntrenamiento.</li></ul>
CU-011	<ul style="list-style-type: none"><li>• InterfazEntrenamiento</li></ul>
CU-012	<ul style="list-style-type: none"><li>• InterfazEntrenamiento</li><li>• GestorFicheroRed</li></ul>
CU-013	<ul style="list-style-type: none"><li>• InterfazEntrenamiento</li><li>• InterfazTratamiento.</li></ul>
CU-014	<ul style="list-style-type: none"><li>• InterfazEntrenamiento</li><li>• InterfazInfoEntrenamiento</li><li>• GestorEvaluacionResumen</li></ul>
CU-015	<ul style="list-style-type: none"><li>• InterfazEntrenamiento</li><li>• GestorEvaluacionResumen</li></ul>
CU-016	<ul style="list-style-type: none"><li>• InterfazEntrenamiento</li><li>• GestorEvaluacionResumen</li></ul>

Tabla 75. Clases asociadas a Casos de uso

### 5.5.2 Descripción de la interacción de objetos

A continuación se representa la interacción descrita en el apartado anterior entre los casos de uso y las clases.

Se han utilizado diagramas de secuencia de manera que se pueda observar con una mayor claridad la interacción existente entre las clases que formarán el sistema.

Cabe destacar que en este apartado se muestran los diagramas de secuencia para los casos de uso más relevantes

A continuación se muestra el diagrama de secuencia para el caso de uso CU-006: Generar ficheros.

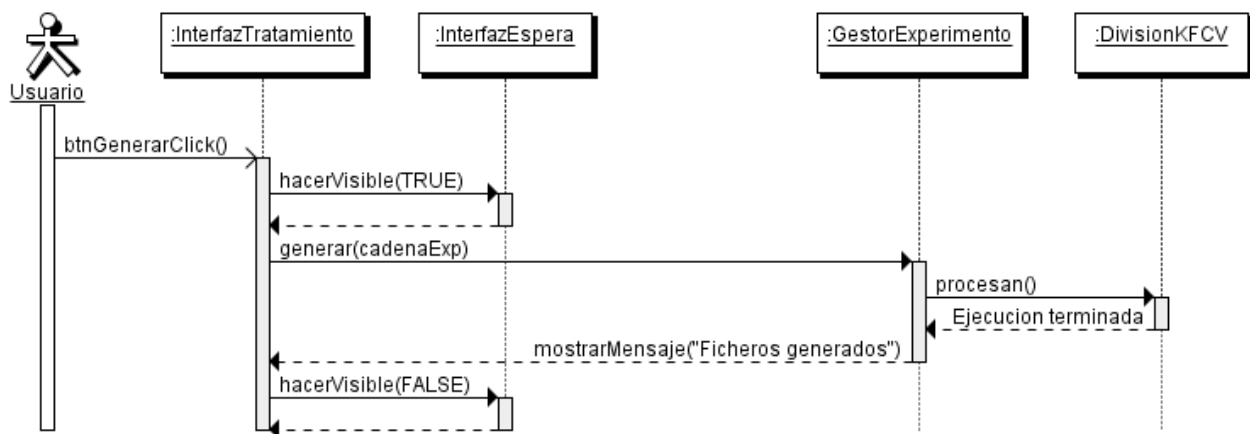


Figura 18. Diagrama de secuencia Caso de Uso CU-006

El siguiente diagrama de secuencia representa el caso de uso CU-014:  
Entrenar

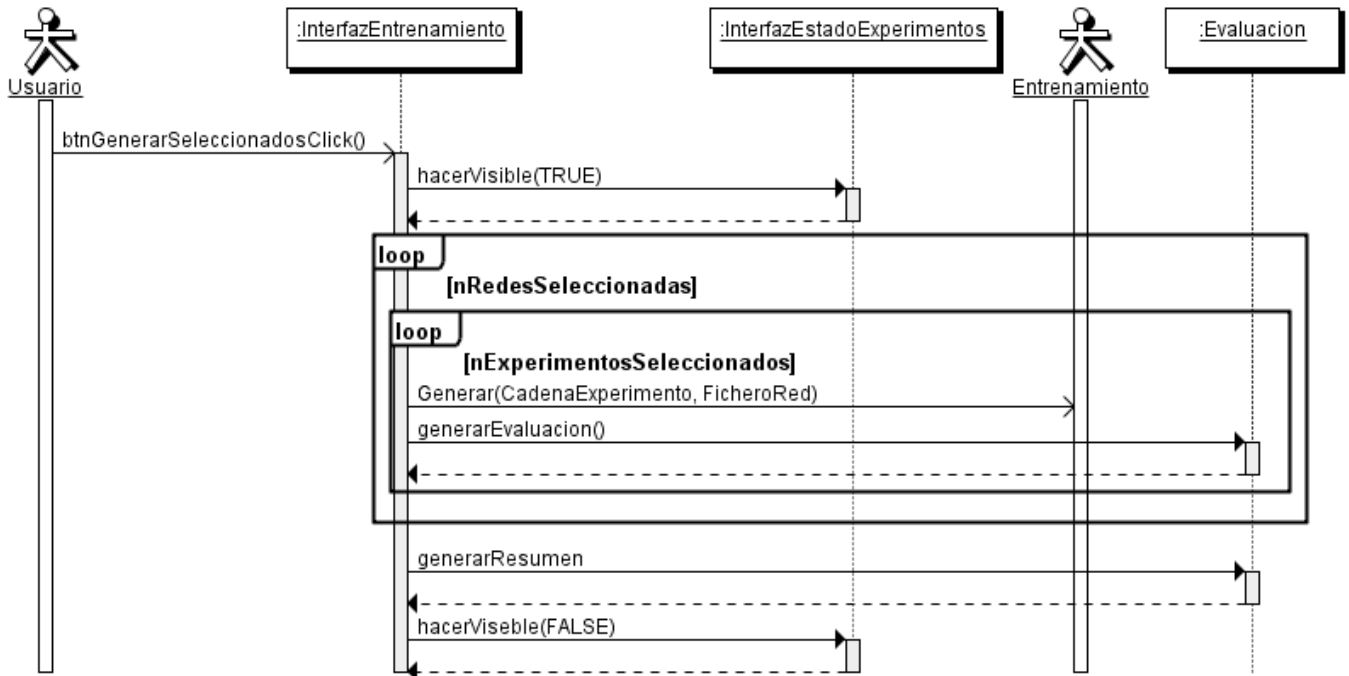


Figura 19. Diagrama de secuencia caso de uso CU-014

## 5.6 Análisis de clases

El objetivo de esta tarea es la descripción de cada una de las clases identificadas en el proceso de análisis. Para ello, se detallarán sus atributos y las relaciones existentes entre ellas.

Es importante destacar, que en esta fase de desarrollo del proyecto, el proceso de análisis, únicamente se realizará un estudio general de los requisitos sin hacer un especial énfasis en cuanto a la toma de decisiones relativas a la implementación. Por lo tanto, el modelo obtenido en este apartado será un simple análisis conceptual de lo que el cliente requiere, una representación a alto nivel del sistema sin ahondar en ningún tipo de detalle relacionado con la implementación. En la siguiente etapa del proyecto, la fase de diseño, se realizará un estudio más exhaustivo.

Además, con el fin de simplificar el diagrama se han incluido las funciones y procedimientos necesarios para la comprensión del sistema, obviando algunas

funciones de obtención o modificación de propiedades, que aunque si serán implementadas no se muestran en este modelo.

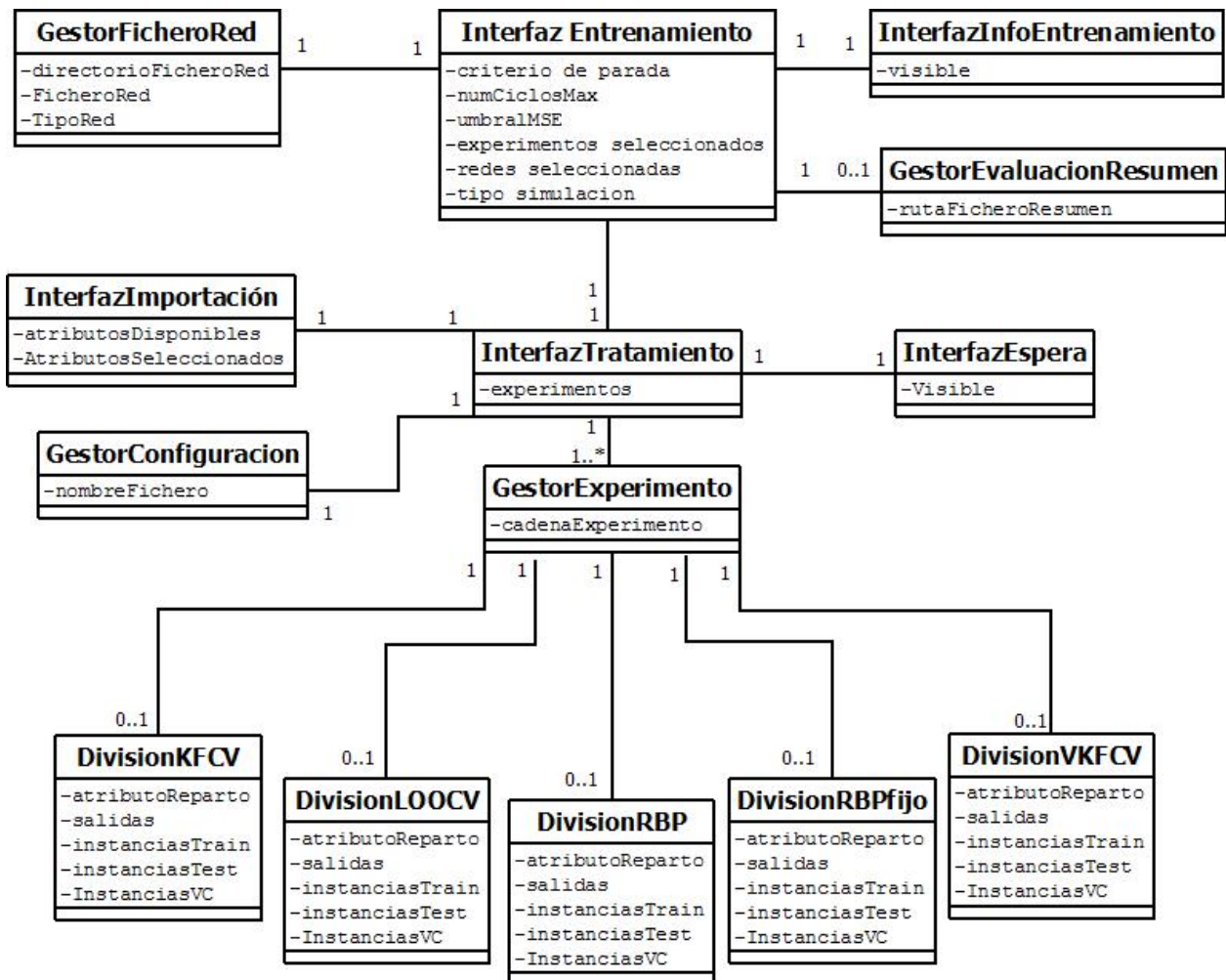


Figura 20. Diagrama conceptual de clases

### 5.6.1 Identificación de responsabilidades y atributos

En este apartado se especifican tanto las responsabilidades asociadas a cada una de las clases mostradas en la ilustración anterior. Con ello, se pretende describir el papel que desempeñan los objetos pertenecientes a las clases en los diferentes casos de uso del sistema.

Además, se describen los atributos correspondientes a las clases, los cuales especifican propiedades de las mismas y están involucrados, a su vez, en las diferentes responsabilidades de la clase.



A continuación se describen las responsabilidades y atributos más relevantes de cada una de las clases del diagrama:

Clase	InterfazTratamiento
Responsabilidades	Clase que gestiona el funcionamiento de la interfaz de tratamiento de ficheros de la aplicación.
Atributos	<b>experimentos:</b> Listado de los experimentos a preparar.

Tabla 76. Clase InterfazTratamiento

Clase	InterfazImportación
Responsabilidades	Clase que gestiona el funcionamiento de la interfaz de usuario que permite al usuario importar datos desde el fichero de origen.
Atributos	<b>atributosDisponibles:</b> Listado de atributos existentes en el fichero de entrada. <b>atributosSeleccionados:</b> Listado de atributos seleccionados por el usuario.

Tabla 77. Clase InterfazImportación

Clase	InterfazEspera
Responsabilidades	Clase que gestiona el funcionamiento de la interfaz mostrada durante la generación de archivos.
Atributos	<b>visible:</b> Estado de la visibilidad de la interfaz.

Tabla 78. Clase InterfazEspera

Clase	GestorConfiguracion
Responsabilidades	Clase encargada de gestionar los archivos de configuración de la aplicación.
Atributos	<b>nombreFichero:</b> Nombre del fichero de configuración a guardar o cargar.

Tabla 79. Clase GestorConfiguracion

Clase	GestorExperimento
Responsabilidades	Clase encargada de la preparación de un experimento añadido por el usuario.
Atributos	<b>cadenaExperimento:</b> Cadena textual identificador del experimento.

Tabla 80. Clase GestorExperimento



Clase	DivisionKFCV
Responsabilidades	Clase encargada de procesar los datos utilizando la técnica de división KFCV.
Atributos	<b>atributoReparto:</b> Atributo en función del cual se deben repartir las instancias. <b>salidas:</b> Atributos seleccionados por el usuario para su predicción. <b>instanciasTrain:</b> Número de instancias para entrenamiento. <b>instanciasTest:</b> Número de instancias para validación. <b>instanciasVC:</b> Número de instancias para validación cruzada.

Tabla 81. Clase DivisionKFCV

Clase	DivisionLOOCV
Responsabilidades	Clase encargada de procesar los datos utilizando la técnica de división LOOCV.
Atributos	<b>atributoReparto:</b> Atributo en función del cual se deben repartir las instancias. <b>salidas:</b> Atributos seleccionados por el usuario para su predicción. <b>instanciasTrain:</b> Número de instancias para entrenamiento. <b>instanciasTest:</b> Número de instancias para validación. <b>instanciasVC:</b> Número de instancias para validación cruzada.

Tabla 82. Clase DivisionLOOCV

Clase	DivisionRBP
Responsabilidades	Clase encargada de procesar los datos utilizando la técnica de división RBP.
Atributos	<b>atributoReparto:</b> Atributo en función del cual se deben repartir las instancias. <b>salidas:</b> Atributos seleccionados por el usuario para su predicción. <b>instanciasTrain:</b> Número de instancias para entrenamiento. <b>instanciasTest:</b> Número de instancias para validación. <b>instanciasVC:</b> Número de instancias para validación cruzada.

Tabla 83. Clase DivisionRBP



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Clase	DivisionRBPfijo
Responsabilidades	Clase encargada de procesar los datos utilizando la técnica de división RBP fijo.
Atributos	<b>atributoReparto:</b> Atributo en función del cual se deben repartir las instancias. <b>salidas:</b> Atributos seleccionados por el usuario para su predicción. <b>instanciasTrain:</b> Número de instancias para entrenamiento. <b>instanciasTest:</b> Número de instancias para validación. <b>instanciasVC:</b> Número de instancias para validación cruzada.

Tabla 84. Clase DivisionRBPfijo

Clase	DivisionVKFCV
Responsabilidades	Clase encargada de procesar los datos utilizando la técnica de división VKFCV.
Atributos	<b>atributoReparto:</b> Atributo en función del cual se deben repartir las instancias. <b>salidas:</b> Atributos seleccionados por el usuario para su predicción. <b>instanciasTrain:</b> Número de instancias para entrenamiento. <b>instanciasTest:</b> Número de instancias para validación. <b>instanciasVC:</b> Número de instancias para validación cruzada.

Tabla 85. Clase DivisionVKFCV





Clase	InterfazEntrenamiento
Responsabilidades	Clase que gestiona el funcionamiento de la interfaz de entrenamiento de experimentos de la aplicación.
Atributos	<b>criterioDeParada:</b> Criterio de parada a utilizar durante el entrenamiento dentro de cada una de las redes seleccionadas. <b>numCiclosMax:</b> Número de ciclos de entrenamiento máximos. <b>umbralMSE:</b> Valor del umbral a utilizar con el criterio de parada por error MSE. <b>experimentosSeleccionados:</b> Lista de los experimentos seleccionados para el entrenamiento. <b>redesSeleccionadas:</b> Lista de las redes seleccionadas para el entrenamiento. <b>tipoSimulacion:</b> valor que indica si la simulación es de clasificación o regresión.

Tabla 86. Clase InterfazEntrenamiento

Clase	GestorFicheroRed
Responsabilidades	Clase que gestiona la ubicación e identificación del tipo de las redes disponibles.
Atributos	<b>directorioFicheroRed:</b> Ruta en la que se encuentran los ficheros de red de neuronas <b>ficheroRed:</b> lista de ficheros de red encontrados en la ruta especificada. <b>tipoRed:</b> tipología de los ficheros de red identificados.

Tabla 87. Clase GestorFicheroRed

Clase	InterfazInfoEntrenamiento
Responsabilidades	Clase que gestiona el funcionamiento de la interfaz mostrada durante el entrenamiento.
Atributos	<b>Visible:</b> estado de la visibilidad e la interfaz

Tabla 88. Clase InterfazInfoEntrenamiento

Clase	GestorEvaluacionResumen
Responsabilidades	Clase que gestiona la generación del fichero de evaluación y resumen del entrenamiento.
Atributos	<b>rutaFicheroResumen:</b> ruta que indica la ubicación del fichero resumen.

Tabla 89. Clase GestorEvaluacionResumen

## 5.6.2 Identificación de asociaciones

En esta tarea se estudian las relaciones de asociación entre las distintas clases identificadas en el diagrama. Estas relaciones han sido obtenidas partir de la especificación de casos de uso.

Cada una de estas asociaciones será especificada con su cardinalidad, número de instancias que pueden estar relacionadas con una instancia del otro lado de la asociación, y con una breve descripción de su funcionalidad.

A continuación se describen las asociaciones identificadas en los diagramas de clases anteriores:

Clases Asociadas	InterfazTratamiento - InterfazImportación
Cardinalidades	InterfazTratamiento (1) InterfazImportación (1)
Descripción	Una interfaz tratamiento utiliza una interfaz de importación para extraer los datos del fichero de origen.

Tabla 90. Asociación clases InterfazTratamiento e InterfazImportación.

Clases Asociadas	InterfazTratamiento - InterfazEspera
Cardinalidades	InterfazTratamiento (1) InterfazEspera (1)
Descripción	Una interfaz tratamiento utiliza una interfaz de espera para mostrarla al generar los archivos.

Tabla 91. Asociación clases InterfazTratamiento e InterfazEspera.

Clases Asociadas	InterfazTratamiento – GestorConfiguración
Cardinalidades	InterfazTratamiento (1) GestorConfiguración (0..1)
Descripción	Una interfaz tratamiento puede utilizar un gestor de configuración para guardar y/o cargar ficheros de configuración.

Tabla 92. Asociación clases InterfazTratamiento y GestorConfiguración.



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Clases Asociadas	InterfazTratamiento - GestorExperimento
Cardinalidades	InterfazTratamiento (1) GestorExperimento (1..*)
Descripción	Una interfaz tratamiento utiliza un gestor de experimento por cada experimento que tenga que preparar la aplicación al generar ficheros.

Tabla 93. Asociación clases InterfazTratamiento y GestorExperimento.

Clases Asociadas	GestorExperimento - DivisionLOOCV
Cardinalidades	GestorExperimento (1) DivisionLOOCV (0..1)
Descripción	Un gestor de experimento puede utilizar o no un objeto de división LOOCV.

Tabla 94. Asociación clases GestorExperimento e DivisionLOOCV.

Clases Asociadas	GestorExperimento - DivisionRBP
Cardinalidades	GestorExperimento (1) DivisionRBP (0..1)
Descripción	Un gestor de experimento puede utilizar o no un objeto de división RBP.

Tabla 95. Asociación clases GestorExperimento e DivisionRBP.

Clases Asociadas	GestorExperimento - DivisionRBPfijo
Cardinalidades	GestorExperimento (1) DivisionRBPfijo (0..1)
Descripción	Un gestor de experimento puede utilizar o no un objeto de división RBP.

Tabla 96. Asociación clases GestorExperimento e DivisionKFCVfijo.

Clases Asociadas	GestorExperimento - DivisionVKFCV
Cardinalidades	GestorExperimento (1) DivisionVKFCV (0..1)
Descripción	Un gestor de experimento puede utilizar o no un objeto de división VKFCV.

Tabla 97. Asociación clases GestorExperimento e DivisionKFCV.



Clases Asociadas	InterfazTratamiento - InterfazEntrenamiento
Cardinalidades	InterfazTratamiento (1) InterfazEntrenamiento (1)
Descripción	Una interfaz tratamiento utiliza una interfaz de Entrenamiento para entrenar los experimentos generados.

Tabla 98. Asociación clases InterfazTratamiento e InterfazEntrenamiento.

Clases Asociadas	InterfazEntrenamiento - GestorFicheroRed
Cardinalidades	InterfazEntrenamiento (1) GestorFicheroRed (1)
Descripción	Una interfaz Entrenamiento utiliza gestor Fichero Red para extraer e identificar las redes disponibles para el entrenamiento

Tabla 99. Asociación clases InterfazEntrenamiento y GestorFicheroRed.

Clases Asociadas	InterfazEntrenamiento - InterfazInfoEntrenamiento
Cardinalidades	InterfazEntrenamiento (1) InterfazInfoEntrenamiento (1)
Descripción	Una interfaz Entrenamiento utiliza una interfaz info entrenamiento para mostrarla durante el entrenamiento.

Tabla 100. Asociación clases InterfazEntrenamiento e InterfazInfoEntrenamiento

Clases Asociadas	InterfazEntrenamiento - GestorEvaluacionResumen
Cardinalidades	InterfazEntrenamiento (1) Interfaz GestorEvaluacionResumen (0..1)
Descripción	Una interfaz Entrenamiento puede utilizar o no un Gestor Evaluación y Resumen para generar las evaluaciones y resumen del entrenamiento.

Tabla 101. Asociación clases InterfazEntrenamiento e InterfazImportación.



### **5.6.3 Identificación de generalizaciones**

En este apartado se estudia otro tipo de relación entre clases, la generalización, donde una clase comparte estructura y/o comportamiento con una o más clases. El término superclase se refiere a la clase que guarda la información común, mientras que el término subclase se refiere a cada uno de los descendientes de la superclase.

En la aplicación no aparece ninguna generalización.



## **5.7 Definición de las interfaces de usuario**

### **5.7.1 Especificación de principios generales de la interfaz**

Uno de los aspectos más importantes de una aplicación con la que interactúan los usuarios es la interfaz, ya que es el medio de comunicación entre el usuario y el sistema. En este proyecto, se contará con una interfaz de usuario.

A pesar de que el perfil de usuario que utilice ambas aplicaciones será un usuario con ciertos conocimientos informáticos, es recomendable que la interfaz de usuario sea lo más clara e intuitiva posible. De esta forma se agilizará el proceso de adaptación de los usuarios al manejo de las aplicaciones.

Es importante además hacer las interfaces los más modulables y reutilizables posible de cara a las posibles actualizaciones que puedan desarrollarse en un futuro.

Con el objetivo de facilitar el proceso de adaptación de los usuarios a las aplicaciones, además de ser interfaces sencillas, se tratará de proporcionar información precisa acerca de posibles errores cometidos en el uso de la interfaz de usuario.

### **5.7.2 Identificación de perfiles y diálogos**

Para la aplicación existirá un único perfil de usuario, el cual podrá acceder a todas las funcionalidades especificadas por los requisitos.

### **5.7.3 Especificación de formatos individuales de la interfaz de la pantalla**

A continuación se muestran las interfaces de usuario para la aplicación, describiendo los elementos que las componen.

En primer lugar, la interfaz tratamiento se muestra a continuación:

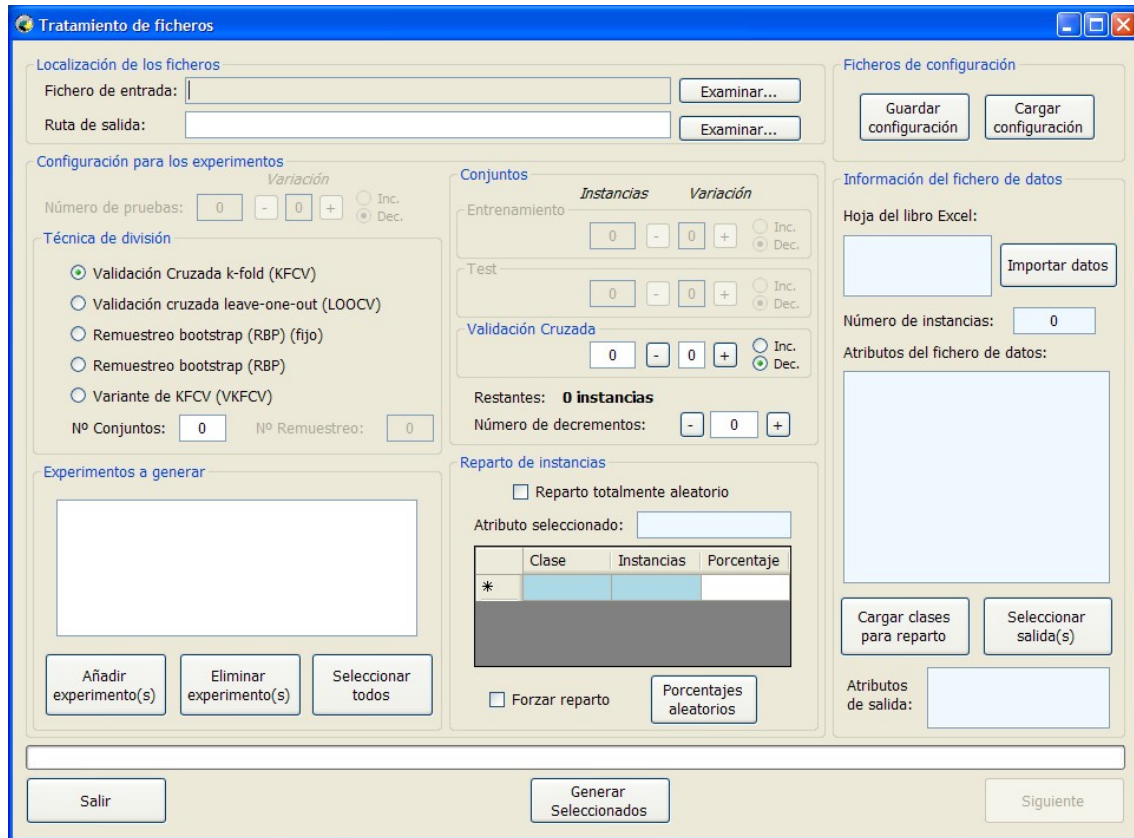


Figura 21. Interfaz Tratamiento

Tal y como se puede observar en la imagen anterior, la interfaz de la aplicación se compone principalmente de los siguientes elementos:

- **Cuadro de texto “Fichero de entrada”:** Permite introducir la ruta en la que se encuentra el fichero de datos original.
- **Cuadro de texto “Ruta de salida”:** Permite especificar la ruta en la que la aplicación generará los ficheros.
- **Botón “Guardar configuración”:** Almacena la configuración de la aplicación en un fichero externo.
- **Botón “Cargar configuración”:** Carga la configuración de la aplicación desde un fichero externo almacenado previamente.
- **Lista “Experimentos a preparar”:** Listado de los experimentos para los que la aplicación debe generar ficheros de entrada.



- **Botón “Añadir experimento(s)”**: Añade los experimentos configurados a la lista “Experimentos a preparar”.
- **Botón “Eliminar experimento(s)”**: Elimina los experimentos seleccionados de la lista “Experimentos a preparar”.
- **Botón “Seleccionar todos”**: selecciona todos los experimentos de la lista “Experimentos a preparar”.
- **Botón “Porcentajes aleatorios”**: Asigna de manera aleatoria porcentajes a las clases cargadas para un atributo determinado.
- **Lista “Hoja del libro Excel”**: Listado de las hojas existentes en el fichero de entrada seleccionado.
- **Botón “Importar datos”**: Abre la interfaz de importación de datos.
- **Lista “Atributos del fichero de datos”**: Muestra los atributos seleccionados por el usuario para incluir en sus ficheros de datos.
- **Botón “Cargar clases para reparto”**: Carga las clases existentes en el fichero de datos para un determinado atributo.
- **Botón “Seleccionar salida(s)”**: Añade los atributos seleccionados a la lista “Atributos de salida”.
- **Lista “Atributos de salida”**: Muestra los atributos seleccionados por el usuario sobre los que se efectuarán las predicciones.
- **Botón “Generar Seleccionados”**: Genera los ficheros para los experimentos seleccionados de la lista “Experimentos a preparar”.
- **Botón “Salir”**: Cierra la aplicación.
- **Botón “Siguiente”**: abre la interfaz de entrenamiento.



La siguiente imagen muestra la interfaz de importación de datos:

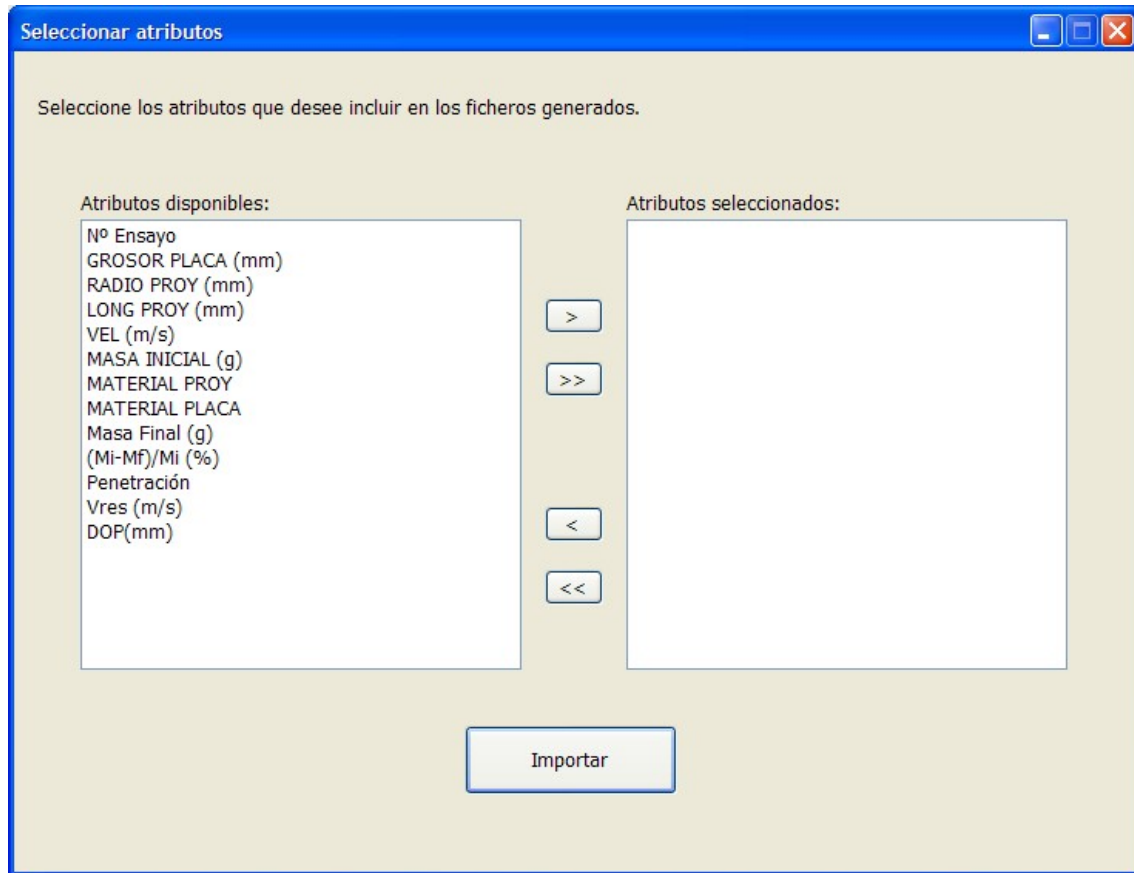


Figura 22. Interfaz de importación

En esta interfaz destacan los siguientes elementos principalmente:

- **Lista "Atributos disponibles":** Listado de todos los atributos existentes en el fichero de datos seleccionado.
- **Lista "Atributos seleccionados":** Listado de los atributos seleccionados por el usuario para su inclusión en los ficheros generados.
- **Botón "Importar":** Importa la información de los atributos incluidos en la lista "Atributos seleccionados".

La siguiente imagen corresponde a la interfaz de entrenamiento:

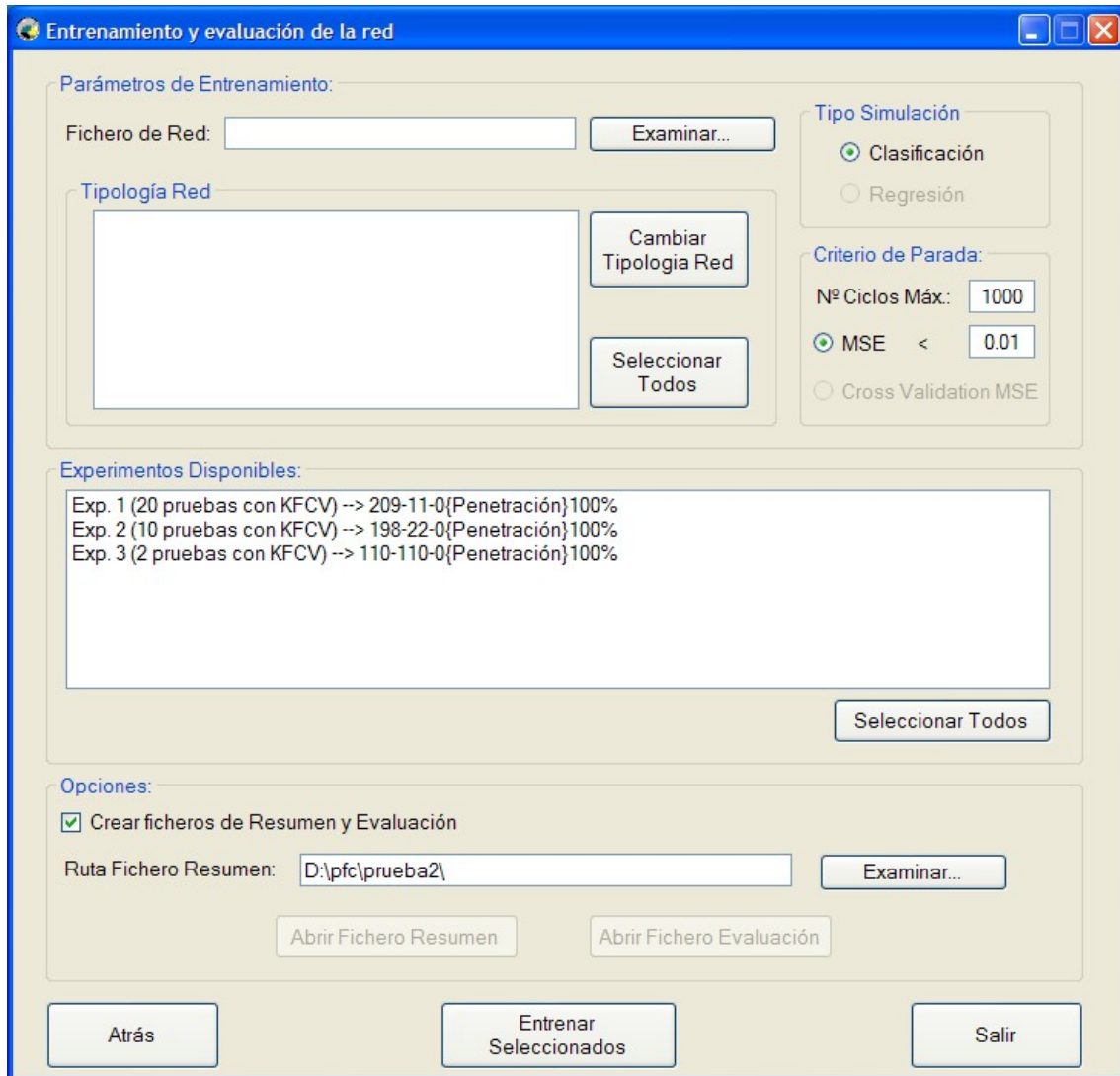


Figura 23. Interfaz de entrenamiento

En esta interfaz destacan los siguientes elementos principalmente:

- **Cuadro de texto “Fichero de Red”:** Permite introducir la ruta del directorio en el que se encuentran los ficheros de red.
- **Botón “Cambiar Tipología Red”:** Abre la interfaz del gestor de los ficheros de red.
- **Botón “Seleccionar Todos” de tipología de red:** selecciona todos los elementos de la lista “Ficheros de red y tipología”.

- **Lista “Ficheros de red y tipología”:** Listado de todos los ficheros de red identificados en el directorio especificado.
- **Lista “Experimentos disponibles”:** Listado de los experimentos disponibles para el entrenamiento.
- **Botón “Seleccionar Todos” de experimentos disponibles:** selecciona todos los elementos de la lista “Experimentos disponibles”.
- **Cuadro de texto “Ruta Fichero Resumen”:** permite introducir la ruta en la que se ubicará el fichero de resumen.
- **Botón “Abrir Fichero Resumen”:** abre el fichero de resumen del entrenamiento.
- **Botón “Abrir Fichero Evaluación”:** abre el fichero de evaluación del entrenamiento del elemento seleccionado de la lista de “Experimentos Disponibles”.
- **Botón “Atrás”:** muestra la interfaz de tratamiento.
- **Botón “Entrenar Seleccionados”:** Entrena los experimentos seleccionados con las redes seleccionadas.
- **Botón “Salir”:** cierra la aplicación.

La siguiente imagen corresponde a la Interfaz de gestor de los ficheros de red



Figura 24. Interfaz gestor de los ficheros de red



Se pueden observar en la anterior imagen los siguientes elementos:

- Lista “Tipologías redes disponibles”: listado de las tipologías disponibles para un fichero de red.
- Botón “Aceptar”: Asigna la tipología al fichero de red seleccionado y muestra la interfaz de entrenamiento.
- Botón “Cancelar”: muestra la interfaz de entrenamiento sin hacer ningún cambio.



## **5.8 *Análisis de consistencia y especificación de requisitos***

En esta sección se detalla cómo se asegurará la calidad entre los distintos modelos que se han generado como productos, comprobando la consistencia entre ellos.

### **5.8.1 Verificación y análisis de consistencia entre modelos**

Mediante esta tarea se comprobará que cada modelo generado es correcto, comprobando la coherencia de los modelos y la existencia de ambigüedad o redundancia.

Se ha comprobado que los casos de uso se pueden realizar con las clases definidas en el modelo de clases.

Por último se muestran las matrices de trazabilidad que permiten asegurar la consistencia existente entre el Estudio de Viabilidad del Sistema y el Análisis.



### 5.8.1.1 Matriz de Requisitos de usuario de capacidad-Casos de uso

	CU-001	CU-002	CU-003	CU-004	CU-005	CU-006	CU-007	CU-008	CU-009	CU-010	CU-011	CU-012	CU-013	CU-014	CU-015	CU-016
RU-C-001	X															
RU-C-002		X														
RU-C-003		X														
RU-C-004			X													
RU-C-005			X													
RU-C-006				X												
RU-C-007				X												
RU-C-008					X											
RU-C-009							X	X								
RU-C-010						X										
RU-C-011									X							
RU-C-012												X				
RU-C-013												X				
RU-C-014											X					
RU-C-015														X		
RU-C-016															X	
RU-C-017																X
RU-C-018										X			X			

Tabla 102. Matriz de trazabilidad Requisitos de Usuario - Casos de Uso

### 5.8.1.2 Matrices de Requisitos software-Casos de uso

	CU-001	CU-002	CU-003	CU-004	CU-005	CU-006	CU-007	CU-008	CU-009	CU-010	CU-011	CU-012	CU-013	CU-014	CU-015	CU-016
RS-F-001	X															
RS-F-002		X														
RS-F-003			X													
RS-F-004		X														
RS-F-005			X													
RS-F-006			X													
RS-F-007			X													
RS-F-008			X													
RS-F-009			X													
RS-F-010				X												
RS-F-011				X												
RS-F-012					X											
RS-F-013							X									
RS-F-014								X								
RS-F-015						X										
RS-F-016									X							
RS-F-017												X				
RS-F-018												X				
RS-F-019											X					
RS-F-020														X		
RS-F-021															X	
RS-F-022																X

Tabla 103. Matriz de trazabilidad Requisitos Software - Casos de Uso

### 5.8.2 Validación de los modelos

Se ha comprobado la integridad de las matrices de trazabilidad asegurando que los modelos planteados son consistentes.



## **6 Diseño del Sistema**

### **6.1 Introducción**

#### **6.1.1 Propósito**

El objetivo principal de este documento es presentar el diseño del sistema realizado de forma detallada. También se estudiará la tecnología que será de utilidad para llevar a cabo esta actividad.

Se realizará una especificación detallada de los componentes en los que se dividirá el sistema, con el fin de cubrir todas las decisiones de diseño correspondientes a la fase de construcción. De esta manera, sólo quedarán posibles decisiones a tomar relacionadas con el lenguaje de programación seleccionado para la codificación del sistema.

El Documento de Diseño del Sistema es de vital importancia en el desarrollo de un proyecto software, ya que marca las pautas para las posteriores fases de construcción y de implantación final del sistema, siendo un documento básico para los programadores encargados de la implementación.



## **6.2 Alcance**

El presente documento presenta una especificación detallada de los componentes en los que se dividirá el sistema, permitiendo cubrir todas las decisiones de diseño que se verán reflejadas en la fase construcción.

Por otro lado, se especificará el entorno tecnológico necesario para que el sistema pueda entrar en ejecución. Contendrá además la planificación de capacidades, los requisitos de administración, el control de accesos, la seguridad y la operación.

A continuación, se describen con mayor detalle las principales tareas que se presentan en este documento:

- **Definición de la arquitectura del sistema:** Se realizará una división del sistema con el objetivo de organizar y facilitar el diseño. Estos subsistemas serán partes lógicas y coherentes, que tendrán interfaces de comunicación entre ellas bien definidas. Una vez definidos estos subsistemas, habrá que acoplarlos adecuadamente dentro de la arquitectura del sistema.
  - **Diseño de la arquitectura de soporte:** En esta tarea se determinarán los mecanismos genéricos de diseño que servirán posteriormente de ayuda para el diseño detallado del sistema de información. Por tanto, se deberá realizar en paralelo con el diseño detallado, ya que hay mucho en común entre ambas actividades y los cambios de una tarea afectan a la otra.
1. **Diseño de casos de uso reales:** Se partirá de los casos de uso identificados en la fase de análisis del sistema y se relacionarán con las clases identificadas en el diseño, en lugar de trabajar con el modelo conceptual del documento de Análisis.
  2. **Diseño de clases:** El objetivo es transformar el modelo de clases del análisis en un modelo de clases de diseño. Dicho modelo es mucho más extenso que el anterior, ya que recoge información de los atributos, métodos y las relaciones que existen entre las distintas clases. Para ello, es fundamental considerar las especificaciones tecnológicas elegidas para la implementación.



3. **Diseño físico de datos:** Indica el formato de los ficheros que la aplicación va a utilizar.
4. **Verificación y aceptación de la arquitectura del sistema:** En este apartado se comprobará la trazabilidad entre componentes, clases y casos de uso.
5. **Especificación técnica del plan de pruebas:** Se especificará en detalle el entorno de pruebas a realizar para comprobar el correcto funcionamiento del sistema.
6. **Establecimiento de requisitos de implantación:** En esta actividad se completará el catálogo de requisitos con aquéllos relacionados con la implantación del sistema en el entorno. Todo esto permite preparar los medios y recursos necesarios para que los usuarios puedan manejar la aplicación sin problemas.

Es importante destacar que el documento que aquí se presenta es una adaptación del diseño planteado por Métrica Versión 3, ya que debido a las características del proyecto (se trata de un proyecto pequeño) algunas de las actividades son demasiado extensas para el problema tratado. Muestra de ello es que la actividad definida por métrica versión 3 DSI 8 Generación de Especificaciones de Construcción no se ha desarrollado. Además, por la misma razón, en otras actividades no se ha profundizado demasiado como es el caso de la actividad DSI 3 Diseño de casos de uso reales en el que se han definido los casos de uso más representativos de cada uno de los módulos del sistema.



## **6.3 Definición de la arquitectura**

### **6.3.1 Definición de los niveles de la arquitectura**

La arquitectura del sistema a desarrollar en el presente proyecto no atiende a ninguno de los modelos estandarizados, utiliza una arquitectura específica en la que interactúa la aplicación con la aplicación de la red de neuronas.

### **6.3.2 Especificación de estándares y normas de diseño y construcción**

Entre las actividades a realizar en este proyecto se encuentra la codificación del sistema, lo cual hace indispensable la tarea de definir un estándar que sirva de guía tanto a diseñadores como a programadores a la hora de generar el código fuente. A continuación se detallan las consideraciones a tener en cuenta:

#### **6.3.2.1 Nombres de Ficheros**

Los nombres de los ficheros que conforman el código fuente del sistema deberán ser representativos de la funcionalidad que contiene cada uno.

Además, su longitud no deberá superar los 25 caracteres y la primera letra deberá estar en escrita en mayúscula. En caso de ser un nombre compuesto la primera letra de cada palabra será escrita en mayúscula y el resto en minúsculas.

Ejemplo: *NombreDeFichero*

#### **6.3.2.2 Idioma**

El idioma utilizado a lo largo de todo el desarrollo del proyecto será el castellano, por lo que los comentarios introducidos en el código fuente deben realizarse en este idioma.



### 6.3.2.3 Cabeceras de las clases

Cada una de las clases generadas mostrará al principio de su código un conjunto de líneas de comentario siguiendo el formato que se muestra a continuación:

```
'Nombre de la clase:  
'Descripción:  
'Autor:
```

### 6.3.2.4 Visibilidad de las clases

Las clases podrán ser públicas (public) o privadas (private).

### 6.3.2.5 Identificadores de las clases

Los identificadores de las clases estarán formados únicamente por letras, como excepción, la letra "ñ" será representada mediante los caracteres "ny".

Además, los identificadores siempre comenzarán con una letra mayúscula. En el caso de que sea un nombre compuesto, la primera letra a partir de la segunda palabra será mayúscula también.

Ejemplo: *IdentificadorCompuesto*

### 6.3.2.6 Constantes

En caso de utilizar constantes, su nombre estará formado únicamente por letras mayúsculas. En el caso de que sea una palabra compuesta, las diferentes palabras se separarán con guiones bajos "\_".

Ejemplo: *CONSTANTE*

### 6.3.2.7 Variables

Los nombres de variables estarán formados por letras minúsculas. En el caso de ser una palabra compuesta, la primera letra de cada palabra posterior a la primera palabra será mayúscula.

Ejemplo: *nombreVariable*



### **6.3.2.8 Funciones y procedimientos**

En cuanto a las funciones y procedimientos, es muy importante que el nombre describa con la mayor precisión posible la funcionalidad que desempeña. Para ello, los nombres utilizados serán verbos que describan la acción que hace la función o procedimiento siempre que sea posible.

Los nombres deberán ser escritos en letras minúsculas únicamente, utilizando mayúsculas en el caso de tratarse de un nombre compuesto, en cuyo caso la primera letra de cada palabra posterior a la primera será escrita en mayúsculas.

Ejemplo: *nombreFuncion*

Los posibles parámetros de entrada o salida de las funciones y procedimientos seguirán el formato de nomenclatura descrito en el apartado 6.3.2.7. Variables

### **6.3.3 Identificación de los subsistemas de la aplicación**

No aplica, el sistema no contiene subsistemas.

### **6.3.4 Especificación del entorno tecnológico**

En este apartado se describe el conjunto de especificaciones referentes al entorno tecnológico del sistema diseñado. Se definirán el conjunto de elementos que compondrán el eje central de la infraestructura técnica que va a adoptar el sistema, además de abordar aquellos inconvenientes técnicos que puedan afectar el desarrollo del proyecto.

#### **6.3.4.1 Hardware**

Para el desarrollo del proyecto se emplearán un sistema: un ordenador un ordenador portátil Compaq.

#### **6.3.4.2 Software**

El sistema operativo instalado en la máquina empleada para la ejecución del sistema es Microsoft Windows XP.

#### **6.3.4.3 Comunicaciones**

No aplica, el sistema no realiza comunicaciones.



### **6.3.5 Especificación de requisitos de seguridad y operación**

Para analizar la especificación de los requisitos de operación se remite al lector al apartado 5.3.1.3. Requisitos de operación.

### **6.3.6 Estudio de la seguridad requerida en el proceso de diseño del sistema**

Al tratarse de un sistema que no maneja información sensible para el usuario del propio sistema no es necesario supervisar la seguridad de las actividades del proceso de diseño.

Por otro lado, con el objetivo de evitar posibles pérdidas de información se realizará un backup diariamente.

### **6.3.7 Análisis de los riesgos del entorno tecnológico**

No aplica, los riesgos asociados al entorno tecnológico son mínimos debido al software utilizado.



## **6.4 Diseño de la arquitectura de soporte**

En este apartado se presentan los elementos más significativos del sistema: los componentes de la arquitectura.

Como se introdujo en el apartado 6.3. Definición de la arquitectura, el sistema tiene una arquitectura específica compuesta por un componente que se comunica la aplicación externa "Neurosolutions" [9].

A continuación se definen las características de los componentes que constituyen el sistema. Para cada uno de ellos se contará con los siguientes atributos:

- **Identificador:** Será único para cada uno de los componentes. Para ello utilizaremos la siguiente nomenclatura: "C-XXX", donde XXX es el número del componente.
- **Nombre:** Nombre del elemento.
- **Tipo:** Tipo de elemento, en este caso será componente.
- **Propósito:** Fin del componente.
- **Función:** De qué se encarga el componente.
- **Subordinados:** Hace referencia a los elementos que en algún modo heredan algún dato del componente en cuestión.
- **Dependencias:** Componente que utiliza el elemento tratado para realizar sus funciones.
- **Interfaz:** Servicios proporcionados para el resto de elementos.
- **Recursos:** Hace referencia a elementos de tipo hardware que el componente puede necesitar para realizar su función.
- **Referencias:** Documentación necesaria para poder entender el componente.
- **Proceso:** Algoritmos especiales de procesamiento.



- **Datos:** Datos internos del elemento, atributos, relaciones de interés que no se hayan indicado en la descripción, o posibles valores de alguno de los elementos.

Identificador	C-001
Nombre	Redes neuronales
Tipo	Componente
Propósito	Realiza el entrenamiento sobre una serie de parámetros seleccionados y ordenamos en función de un criterio inteligente
Función	Recoge los parámetros de entrada y los trata para poder realizar el entrenamiento en las redes neuronales seleccionadas.
Subordinados	No aplica
Dependencias	No aplica
Interfaz	No aplica
Recursos	No aplica
Referencias	No aplica
Proceso	No aplica
Datos	No aplica

Tabla 104. Componente C-001: Redes neuronales





## **6.5 Diseño de casos de uso reales**

En este apartado se realiza un estudio de los casos de uso identificados en la fase de Análisis, con el fin de identificar las clases de diseño necesarias para los distintos módulos que componen el sistema, así como las relaciones entre ellas.

El objetivo es identificar el mínimo número de clases necesarias para el sistema, siempre y cuando permitan cubrir todos los requisitos especificados anteriormente.

### **6.5.1 Identificación de clases asociadas a un caso de uso**

A continuación se muestran las clases identificadas en el diseño y su asociación con los distintos caso de uso del sistema.

Identificador	Clases Asociadas
CU-001	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-002	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• Interfaz importación.</li></ul>
CU-003	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-004	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-005	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-006	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• GestorExperimento.</li><li>• DivisionKFCV.</li><li>• DivisionLOOCV.</li><li>• DivisionRBP.</li><li>• DivisionVKFCV.</li></ul>
CU-007	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• Gestor Configuración.</li></ul>
CU-008	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• Gestor Configuración.</li></ul>
CU-009	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li></ul>
CU-010	<ul style="list-style-type: none"><li>• InterfazTratamiento.</li><li>• InterfazEntrenamiento.</li></ul>
CU-011	<ul style="list-style-type: none"><li>• InterfazEntrenamiento</li></ul>
CU-012	<ul style="list-style-type: none"><li>• InterfazEntrenamiento</li><li>• GestorFicheroRed</li></ul>

CU-013	<ul style="list-style-type: none"> <li>• InterfazEntrenamiento</li> <li>• InterfazTratamiento.</li> </ul>
CU-014	<ul style="list-style-type: none"> <li>• InterfazEntrenamiento</li> <li>• InterfazInfoEntrenamiento</li> <li>• GestorEvaluacionResumen</li> </ul>
CU-015	<ul style="list-style-type: none"> <li>• InterfazEntrenamiento</li> <li>• GestorEvaluacionResumen</li> </ul>
CU-016	<ul style="list-style-type: none"> <li>• InterfazEntrenamiento</li> <li>• GestorEvaluacionResumen</li> </ul>

Tabla 105. Clases de diseño asociadas a Casos de uso

## 6.5.2 Diseño de la realización de los casos de uso

En este apartado se especifican de forma extendida los casos de uso detallados en el análisis, haciendo corresponder los distintos escenarios con acciones concretas identificadas en las clases obtenidas en el desarrollo del documento.

A continuación se muestra el diagrama de secuencia para el caso de uso CU-006: Generar ficheros.

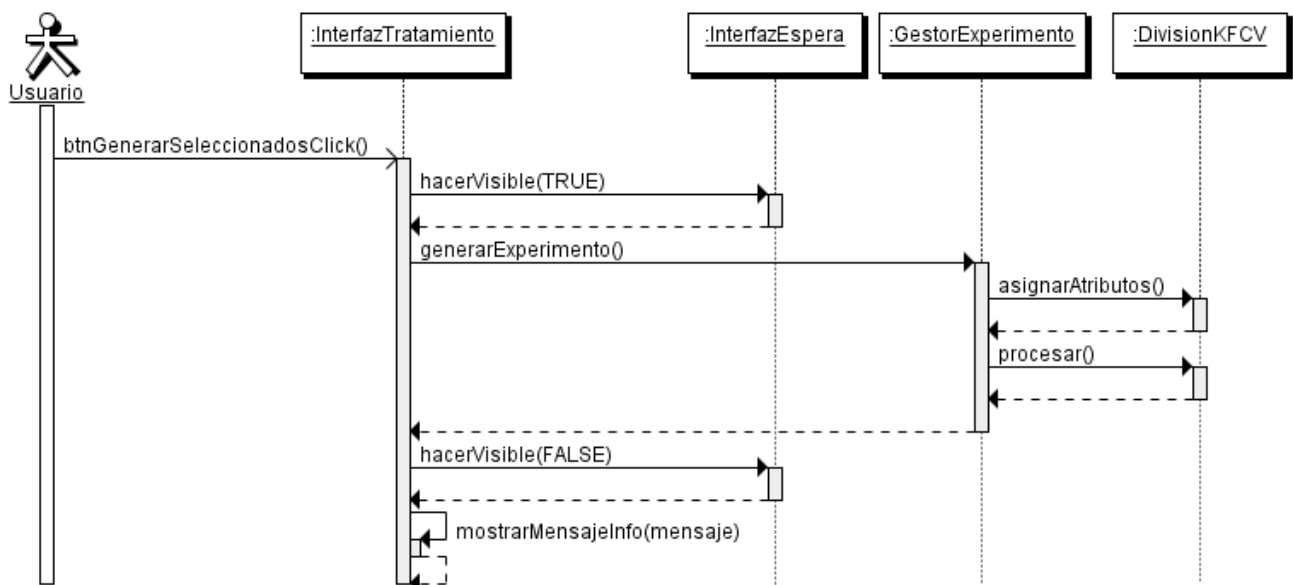


Figura 25. Diagrama de secuencia extendido del caso de uso CU-006

El siguiente diagrama de secuencia representa el caso de uso CU-014:  
Entrenar

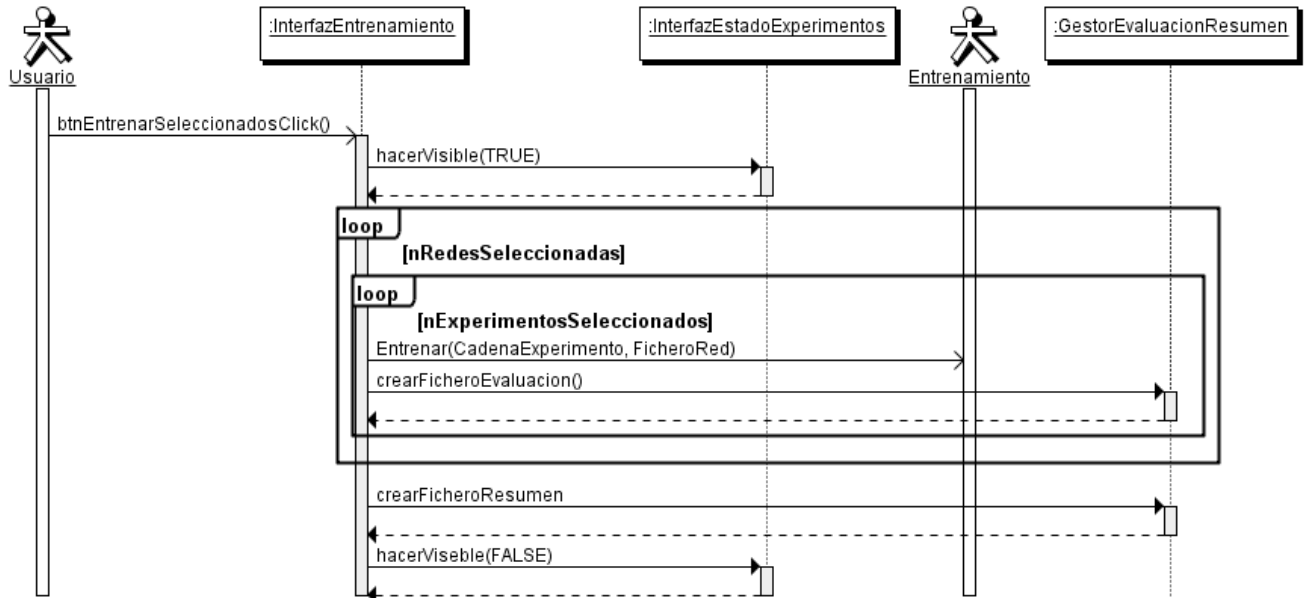


Figura 26. Diagrama de secuencia extendido del caso de uso CU-014



## **6.6 *Diseño de clases***

En esta sección se desarrolla el modelo de clases del sistema que se está diseñando, completando el realizado durante la fase de Análisis del Sistema.

Los modelos aquí presentados servirán de guía al programador, por lo que los identificadores, relaciones, atributos y funciones son orientativos, dejando al programador libertad para programar bajo su propio criterio.

### **6.6.1 Identificación de clases de diseño**

A continuación se muestran los modelos de clases de diseño de la aplicación:

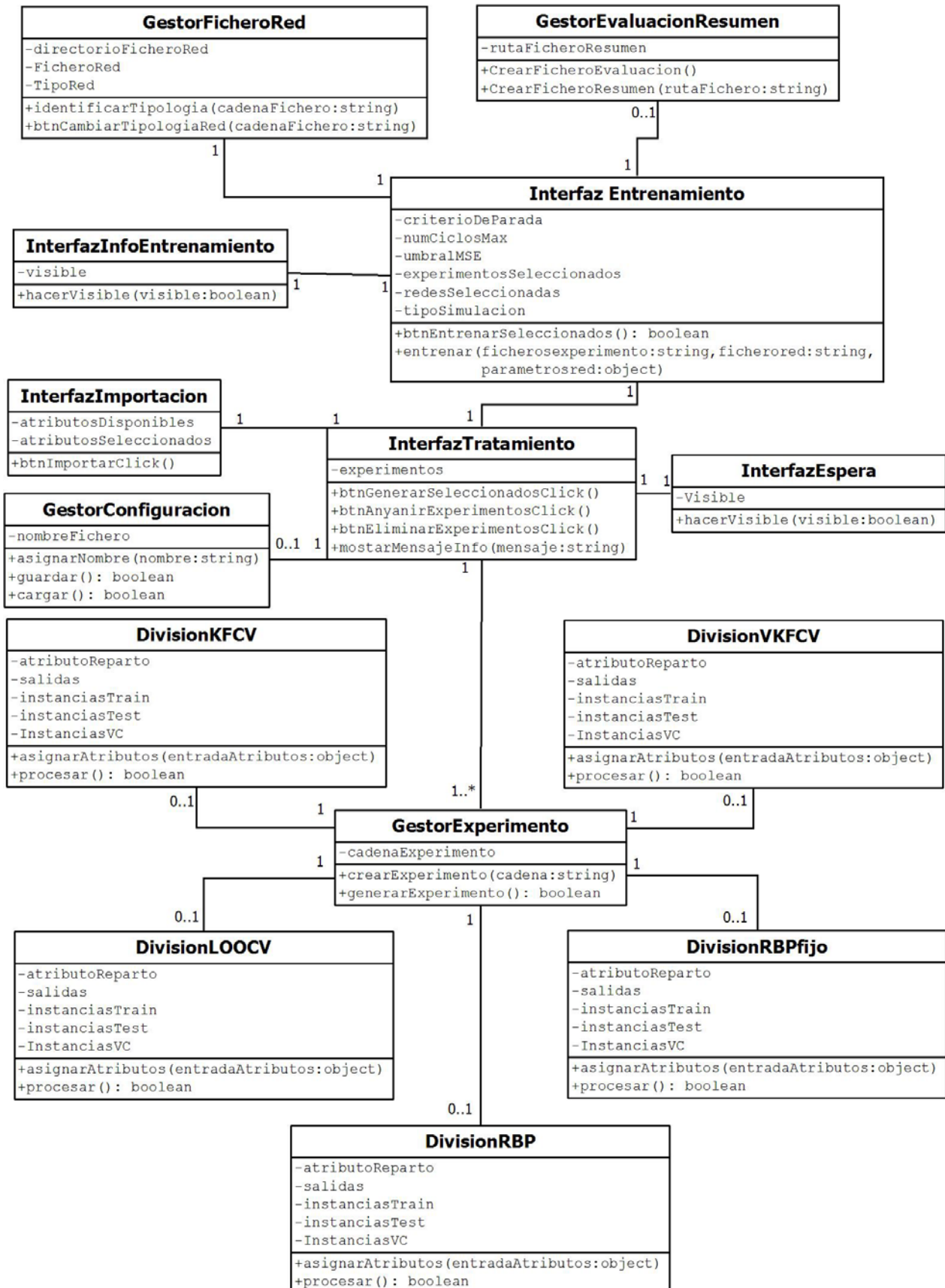


Figura 27. Diagrama de clases de diseño



## 6.6.2 Identificación de atributos y métodos de clases de diseño

A continuación se describen las clases de diseño identificadas, incluyendo el detalle de sus atributos y métodos.

Clase	InterfazTratamiento
Descripción	Clase encargada de gestionar la interfaz de tratamiento de ficheros para interaccionar con el usuario.
Dependencias	InterfazImportación, InterfazEspera, Gestor Experimento, GestorConfiguración, InterfazTratamiento.
ATRIBUTOS	
Identificador	experimentos
Descripción	Listado de experimentos a preparar.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>btnGenerarSeleccionadosClick()</i>
Descripción	Controla la funcionalidad asociada al botón "Generar Seleccionados" de la interfaz.
Signatura	<i>btnAnyadirExperimentosClick()</i>
Descripción	Controla la funcionalidad asociada al botón "Añadir Experimento(s)" de la interfaz.
Signatura	<i>btnEliminarExperimentosClick()</i>
Descripción	Controla la funcionalidad asociada al botón "Eliminar Experimento(s)" de la interfaz.
Signatura	<i>mostrarMensajeInfo(mensaje)</i>
Descripción	Muestra por pantalla un cuadro de diálogo informativo con el mensaje recibido como parámetro.

Tabla 106. Atributos, funciones y procedimientos clase InterfazTratamiento



Clase	InterfazImportacion
Descripción	Clase encargada de gestionar las funcionalidades proporcionadas por la interfaz para la importación de datos.
Dependencias	InterfazTratamiento.
ATRIBUTOS	
Identificador	<i>atributosDisponibles</i>
Descripción	Listado de atributos disponibles en el fichero de datos de entrada.
Identificador	<i>atributosSeleccionados</i>
Descripción	Listado de atributos seleccionados por el usuario.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>btnImportarClick()</i>
Descripción	Controla la funcionalidad asociada al botón "Importar".

Tabla 107. Atributos, funciones y procedimientos clase InterfazImportación

Clase	InterfazEspera
Descripción	Clase encargada de gestionar la interfaz de espera, utilizada cuando la aplicación está generando los archivos para los experimentos.
Dependencias	InterfazTratamiento
ATRIBUTOS	
Identificador	<i>visible</i>
Descripción	Indica si la interfaz es visible o no.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>hacerVisible(visibilidad)</i>
Descripción	Modifica el estado de la visibilidad de la interfaz.

Tabla 108. Atributos, funciones y procedimientos clase InterfazEspera



Clase	GestorConfiguracion
Descripción	Clase encargada de gestionar el almacenamiento y recuperación de los ficheros de configuración.
Dependencias	InterfazTratamiento
ATRIBUTOS	
Identificador	<i>nombreFichero</i>
Descripción	Nombre del fichero de configuración a guardar o cargar.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>asignarNombre(nombre)</i>
Descripción	Asigna el valor del parámetro "nombre" al atributo "nombreFichero".
Signatura	<i>guardar()</i>
Descripción	Almacena la configuración en el archivo con nombre "nombreFichero".
Signatura	<i>cargar()</i>
Descripción	Recupera la configuración del archivo con nombre "nombreFichero".

Tabla 109. Atributos, funciones y procedimientos clase GestorConfiguracion

Clase	GestorExperimento
Descripción	Clase encargada de la generación de un experimento.
Dependencias	InterfazPrincipal, DivisionKFCV, DivisionLOOCV, DivisionRBP, DivisionRBPfijo, DivisionVKFCV.
ATRIBUTOS	
Identificador	<i>cadenaExperimento</i>
Descripción	Cadena de caracteres que representa al experimento a preparar.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>CrearExperimento(cadena)</i>
Descripción	Asigna el valor "cadena" al atributo "cadenaExperimento".
Signatura	<i>generarExperimento()</i>
Descripción	Genera los archivos correspondientes al experimento identificado por su "cadenaExperimento".

Tabla 110. Atributos, funciones y procedimientos clase GestorExperimento





Clase	DivisionKFCV
Descripción	Clase encargada de realizar el reparto de instancias usando la división de tipo KFCV.
Dependencias	GestorExperimento.
ATRIBUTOS	
Identificador	<i>atributoReparto</i>
Descripción	Atributo utilizado para repartir las instancias entre los distintos conjuntos.
Identificador	<i>salidas</i>
Descripción	Conjunto de atributos seleccionados por el usuario como salidas para la predicción del sistema.
Identificador	<i>instanciasTrain</i>
Descripción	Cantidad de instancias requeridas para entrenamiento.
Identificador	<i>instanciasTest</i>
Descripción	Cantidad de instancias requeridas para validación.
Identificador	<i>instanciasVC</i>
Descripción	Cantidad de instancias requeridas para validación cruzada.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>asignarAtributos(atributos)</i>
Descripción	Asigna los atributos de reparto y salida a los atributos de la clase.
Signatura	<i>procesar()</i>
Descripción	Realiza el procesamiento del reparto de instancias.

Tabla 111. Atributos, funciones y procedimientos clase DivisionKFCV



Clase	DivisionLOOCV
Descripción	Clase encargada de realizar el reparto de instancias usando la división de tipo LOOCV.
Dependencias	GestorExperimento.
ATRIBUTOS	
Identificador	<i>atributoReparto</i>
Descripción	Atributo utilizado para repartir las instancias entre los distintos conjuntos.
Identificador	<i>salidas</i>
Descripción	Conjunto de atributos seleccionados por el usuario como salidas para la predicción del sistema.
Identificador	<i>instanciasTrain</i>
Descripción	Cantidad de instancias requeridas para entrenamiento.
Identificador	<i>instanciasTest</i>
Descripción	Cantidad de instancias requeridas para validación.
Identificador	<i>instanciasVC</i>
Descripción	Cantidad de instancias requeridas para validación cruzada.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>asignarAtributos(atributos)</i>
Descripción	Asigna los atributos de reparto y salida a los atributos de la clase.
Signatura	<i>procesar()</i>
Descripción	Realiza el procesamiento del reparto de instancias.

Tabla 112. Atributos, funciones y procedimientos clase DivisionLOOCV



Clase	DivisionRBP
Descripción	Clase encargada de realizar el reparto de instancias usando la división de tipo RBP.
Dependencias	GestorExperimento.
ATRIBUTOS	
Identificador	<i>atributoReparto</i>
Descripción	Atributo utilizado para repartir las instancias entre los distintos conjuntos.
Identificador	<i>salidas</i>
Descripción	Conjunto de atributos seleccionados por el usuario como salidas para la predicción del sistema.
Identificador	<i>instanciasTrain</i>
Descripción	Cantidad de instancias requeridas para entrenamiento.
Identificador	<i>instanciasTest</i>
Descripción	Cantidad de instancias requeridas para validación.
Identificador	<i>instanciasVC</i>
Descripción	Cantidad de instancias requeridas para validación cruzada.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>asignarAtributos(atributos)</i>
Descripción	Asigna los atributos de reparto y salida a los atributos de la clase.
Signatura	<i>procesar()</i>
Descripción	Realiza el procesamiento del reparto de instancias.

Tabla 113. Atributos, funciones y procedimientos clase DivisionRBP



Clase	DivisionRBPfijo
Descripción	Clase encargada de realizar el reparto de instancias usando la división de tipo RBPfijo.
Dependencias	GestorExperimento.
ATRIBUTOS	
Identificador	<i>atributoReparto</i>
Descripción	Atributo utilizado para repartir las instancias entre los distintos conjuntos.
Identificador	<i>salidas</i>
Descripción	Conjunto de atributos seleccionados por el usuario como salidas para la predicción del sistema.
Identificador	<i>instanciasTrain</i>
Descripción	Cantidad de instancias requeridas para entrenamiento.
Identificador	<i>instanciasTest</i>
Descripción	Cantidad de instancias requeridas para validación.
Identificador	<i>instanciasVC</i>
Descripción	Cantidad de instancias requeridas para validación cruzada.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>asignarAtributos(atributos)</i>
Descripción	Asigna los atributos de reparto y salida a los atributos de la clase.
Signatura	<i>procesar()</i>
Descripción	Realiza el procesamiento del reparto de instancias.

Tabla 114. Atributos, funciones y procedimientos clase DivisionRBPfijo



Clase	DivisionVKFCV
Descripción	Clase encargada de realizar el reparto de instancias usando la división de tipo VKFCV.
Dependencias	GestorExperimento.
ATRIBUTOS	
Identificador	<i>atributoReparto</i>
Descripción	Atributo utilizado para repartir las instancias entre los distintos conjuntos.
Identificador	<i>salidas</i>
Descripción	Conjunto de atributos seleccionados por el usuario como salidas para la predicción del sistema.
Identificador	<i>instanciasTrain</i>
Descripción	Cantidad de instancias requeridas para entrenamiento.
Identificador	<i>instanciasTest</i>
Descripción	Cantidad de instancias requeridas para validación.
Identificador	<i>instanciasVC</i>
Descripción	Cantidad de instancias requeridas para validación cruzada.
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>asignarAtributos(atributos)</i>
Descripción	Asigna los atributos de reparto y salida a los atributos de la clase.
Signatura	<i>procesar()</i>
Descripción	Realiza el procesamiento del reparto de instancias.

Tabla 115. Atributos, funciones y procedimientos clase DivisionVKFCV



Clase	InterfazEntrenamiento
Descripción	Clase encargada de gestionar las funcionalidades proporcionadas por la interfaz para el entrenamiento de los experimentos.
Dependencias	InterfazTratamiento, GestorFicherosRed, GestorevaluacionResumen
ATRIBUTOS	
Identificador	<i>criterioDeParada</i>
Descripción	Atributo utilizado para almacenar el criterio de para del entrenamiento en cada red.
Identificador	<i>numCiclosMax</i>
Descripción	Atributo utilizado para indicar el número de ciclos máximo en el entrenamiento.
Identificador	<i>umbralMSE</i>
Descripción	Atributo utilizado para indicar el umbral MSE en el entrenamiento.
Identificador	<i>experimentosSeleccionados</i>
Descripción	Listado de experimentos a entrenar
Identificador	<i>redesSeleccionadas</i>
Descripción	Listado de redes a entrenar
Identificador	<i>tipoSimulacion</i>
Descripción	Atributo utilizado para almacenar el tipo de simulación
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>btnEntrenarSeleccionados</i>
Descripción	Controla la funcionalidad asociada al botón "Entrenar seleccionados" de la interfaz
Signatura	<i>Entrenar</i>
Descripción	Entrena un experimento en una determinada red.

Tabla 116. Atributos, funciones y procedimientos clase InterfazEntrenamiento



Clase	GestorFicheroRed
Descripción	Clase encargada de gestionar la identificación de los ficheros de red
Dependencias	interfazEntrenamiento
ATRIBUTOS	
Identificador	<i>directorioFicheroRed</i>
Descripción	Ruta del directorio donde se encuentran almacenadas las redes
Identificador	<i>ficheroRed</i>
Descripción	Listado de los ficheros de red identificados
Identificador	<i>tipoRed</i>
Descripción	Lista de las tipologías de red disponibles
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>identificarTipologia</i>
Descripción	Identifica la tipología de un fichero de red.
Signatura	<i>btnCambiarTipologiaRed</i>
Descripción	Cambia la tipología de un fichero de red

Tabla 117. Atributos, funciones y procedimientos clase GestorFicheroRed

Clase	GestorEvaluacionResumen
Descripción	Clase encargada de gestionar la creación de los informes de evaluación y resumen del entrenamiento
Dependencias	interfazEntrenamiento
ATRIBUTOS	
Identificador	<i>rutaFicheroResumen</i>
Descripción	ruta del fichero donde se almacena el informe resumen
FUNCIONES Y PROCEDIMIENTOS	
Signatura	<i>crearFicheroEvaluacion</i>
Descripción	Crea el informe de evaluación del entrenamiento
Signatura	<i>crearFicheroResumen</i>
Descripción	Crea el informe de resumen del entrenamiento

Tabla 118. Atributos, funciones y procedimientos clase GestorEvaluacionResumen



## 6.7 Diseño físico de datos

La aplicación, para el almacenamiento de los ficheros de configuración utiliza ficheros en formato Microsoft Excel. En ellos almacena tanto los experimentos de la lista "Experimentos a preparar" de la aplicación como el resto de opciones de configuración.

A continuación se muestra un ejemplo de fichero de configuración almacenado:

	A	B	C	D	E
1	F:\documentos\mikel\pfc\DATOS.xls	Penetración	30	GROSOR PLACA (mm)	Penetración
2	F:\documentos\mikel\pfc\ejemplo\		70	RADIO PROY (mm)	
3	Exp. 1 (6 pruebas con VKFCV) --> 100-60-40{Penetración}30%[no]-70%[si]		0	LONG PROY (mm)	
4			0	VEL (m/s)	
5			0	MASA INICIAL (g)	
6				MATERIAL PROY	
7				MATERIAL PLACA	
8				Penetración	
9					

Figura 28. Ejemplo de fichero de configuración

Para saber mas detalles sobre el formato necesario que debe utilizarse para los ficheros de entrada véase el Anexo B: Formato para ficheros de entrada.





## 6.8 Verificación y aceptación de la arquitectura del sistema

El objetivo de la tarea documentada en este apartado es el de garantizar la calidad y viabilidad de las especificaciones del diseño del sistema de información antes de realizar el diseño detallado.

### 6.8.1 Análisis de consistencia de las especificaciones de diseño

Esta tarea pretende asegurar que las especificaciones de diseño realizadas hasta el momento son coherentes, comprobando que no son ambiguas, que no hay información duplicada y que todos los casos de uso quedan cubiertos con el diseño.

Estas comprobaciones se fundamentan en técnicas matriciales o de revisión entre los elementos comunes de los distintos modelos.

#### 6.8.1.1 Trazabilidad casos de uso – clases de diseño

	CU-001	CU-002	CU-003	CU-004	CU-005	CU-006	CU-007	CU-008	CU-009	CU-010	CU-011	CU-012	CU-013	CU-014	CU-015	CU-016
InterfazTratamiento	X	X	X	X	X	X	X	X	X	X						
InterfazImportacion		X														
InterfazEspera						X										
GestorExperimento						X										
GestorConfiguracion							X	X								
DivisionKFCV						X										
DivisionLOOCV						X										
DivisionRBP						X										
DivisionRBPfijo						X										
DivisionVKFCV						X										
InterfazEntrenamiento									X		X	X	X			
GestorFicheroRed												X		X	X	X
GestorEvaluacionResumen														X	X	X

Tabla 119. Matriz de trazabilidad Clases de diseño - Casos de Uso



## **6.9 Especificación Técnica del Plan de Pruebas**

El objetivo de la tarea descrita en este apartado es el de especificar un plan detallado de las diferentes pruebas a realizar una vez implementado el sistema.

La elaboración de un plan de pruebas detallado tiene como finalidad comprobar que el sistema desarrollado cumple todas las funcionalidades requeridas, no presenta errores y proporciona un nivel de calidad suficiente.

El plan de pruebas se ha realizado tomando como punto de partida el catálogo de requisitos y el diseño detallado el sistema. Se puede analizar el plan de pruebas resultante en el apartado 7. Plan de Validación y Verificación del Software.



## **6.10 Establecimiento de Requisitos de Implantación**

### **6.10.1 Especificación de Requisitos de Implantación**

En este apartado se especifican los requisitos de implantación del proyecto. Estos requisitos están relacionados con la formación, infraestructura e instalación, siendo necesarios para preparar y organizar la implantación del sistema de forma adecuada.

El identificador de cada uno de los requisitos de implantación tendrá el siguiente formato: RS-IMP-XXX, donde XXX representa un valor numérico entre 000 y 999.

A continuación se detallan los requisitos de implantación identificados para el presente proyecto:

Identificador	RS-IMP-001		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Disponer de un ordenador con sistema operativo Microsoft Windows XP		

Tabla 120. Requisito de software RS-IMP-001



Identificador	RS-IMP-002		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Disponer de un ordenador con la aplicación Microsoft office Excel 2003		

Tabla 121. Requisito de software RS-IMP-002

Identificador	RS-IMP-003		
Prioridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Fuente	Cliente
Necesidad	<input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	Estabilidad	Estable
Claridad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad	<input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja
Descripción	Disponer de un ordenador con la aplicación NeuroSolutions 5		

Tabla 122. Requisito de software RS-IMP-003



## **7 Plan de Validación y Verificación del Software**

### **7.1 Introducción**

#### **7.1.1 Propósito del documento**

El objetivo de este documento es presentar la definición de un plan de pruebas que permitan comprobar que el sistema funciona correctamente, cumpliendo con las especificaciones dadas por el cliente en su solicitud.

El plan de pruebas propuesto detalla todas las características a comprobar del sistema, junto con los procedimientos a llevar a cabo para realizar dichas comprobaciones. Cada una de las pruebas a realizar debe llevar asociada una descripción completa, los pasos a seguir para su realización y los criterios para la aceptación de la prueba. Asimismo se especifica también el entorno necesario para la realización de las pruebas, y el proceso de realización de informes de las pruebas realizadas.

Es destacable la importancia de una buena definición de requisitos software y hardware previa, que permita garantizar una correcta evaluación del sistema, haciendo que los resultados de las pruebas realizadas se acerquen en la mayor medida posible a los obtenidos por el sistema en el entorno de explotación.

Teniendo como guía el presente documento, obtendremos un sistema verificado y validado, funcionando correctamente y asegurando la satisfacción del cliente debido el cumplimiento de su solicitud.



## **7.2 Plan de prueba**

Para poder desarrollar un sistema informático de calidad que funcione de manera correcta es imprescindible establecer un buen plan de pruebas.

En este apartado se definen los objetivos de las pruebas a realizar y cómo han de llevarse a cabo. Debe quedar recogido qué elementos van a ser probados, las características del sistema que se van a identificar, los elementos entregables, las actividades a realizar para poder construir una prueba, qué requisitos debe tener el sistema para que las pruebas se ejecuten con éxito y, por último, los criterios a tener en cuenta por el equipo de desarrollo para determinar si una prueba es finalizada con éxito o no.

### **7.2.1 Definición del Alcance de las pruebas**

Para cubrir el objetivo de elaborar un plan de pruebas detallado, se presentará los tipos de pruebas necesarias para validar correctamente la implementación del sistema:

- **Pruebas Unitarias:** con ellas se comprobará el correcto funcionamiento de los componentes y clases presentes en el sistema con el objetivo de asegurar que cada uno de los elementos funcionan correctamente por separado.
- **Pruebas de Integración:** el propósito que persigue estas pruebas es la de comprobar que todas las partes del sistema están ensambladas correctamente y su interacción no produce errores o salidas no esperadas.
- **Pruebas de Aceptación:** se centran en abarcar todas aquellas situaciones que verifican los requisitos software descritos por el cliente y que además son impuestos por el cliente como mínimos para aceptar como válido el sistema.
- **Pruebas del Sistema:** tiene como objetivo verificar el correcto funcionamiento de los componentes del sistema mediante diferentes pruebas de caja negra que cubren diferentes ámbitos (rendimiento, resistencia ante fallos,...)



- **Pruebas de Implantación:** se centran en verificar que una vez se ha implantado el sistema en el cliente, éste funciona correctamente.

Los objetivos de este apartado serán los siguientes:

- Comprobar la completitud de la estructura y funcionalidad de los componentes del sistema.
- Verificar la interacción correcta entre componentes.
- Comprobar el correcto funcionamiento del sistema en el entorno de operación final muy similar al que el cliente puede utilizar.
- Corroborar que el sistema es estable ante diferentes situaciones.
- En caso de posibles errores controlados, ofrecer mensajes que ayuden a identificar cual es el problema facilitándole la labor de conseguir el objetivo correctamente.

### 7.2.2 Entregables

Para poder probar el sistema es necesario tener cierto conocimiento acerca de cómo funciona éste y de los recursos que son necesarios para el mismo, además de saber qué criterios se han definido para que el sistema pueda ser considerado válido y, por tanto, cumpla con las necesidades del cliente.

Por ello, se hace indispensable disponer de una serie de documentos y recursos antes de la realización de las pruebas:

- Documento de Estudio de la Viabilidad.
- Documento de Análisis del Sistema.
- Documento de Diseño del Sistema.
- Plan de Validación y Verificación del Software.
- El software del sistema.
- Manuales de usuario del sistema.



Una vez que se dispone de los recursos necesarios para la realización de las pruebas, se procederá a ejecutar cada una de ellas en el sistema. Los resultados de las pruebas serán recogidos en un informe denominado Informe de realización de pruebas del sistema, en el cual se registrará toda información acerca de las pruebas realizadas. Dicho informe deberá contener una tabla por cada prueba realizada, siguiendo la plantilla definida para tal fin en el apartado 7.4. Plantilla de informe de pruebas.

### **7.2.3 Tareas de prueba**

Desarrollar un sistema de calidad que cumpla con el resultado esperado por el cliente requiere definir un conjunto de pruebas lo más completo posible, de forma que cada una de las funcionalidades del sistema se compruebe correctamente.

Debido a que el conjunto de pruebas puede llegar a ser excesivamente numeroso se hace necesaria la definición de un conjunto de tareas que sirvan de guía para preparar y desarrollar cada una de las pruebas.

Por lo tanto, las tareas definidas para desarrollar las pruebas de este proyecto son las siguientes:

- **Definición de la prueba:** Se debe realizar una descripción acerca de la prueba, especificando cual es el objetivo de la misma, qué elementos del sistema serán sometidos a prueba y las características que debe tener el sistema para que la prueba pueda ser ejecutada. Además se detallará cada uno de los pasos a realizar para la correcta ejecución de la prueba.
- **Ejecución de la prueba:** Observando el comportamiento del sistema y los resultados obtenidos se podrá determinar si la prueba es superada con éxito o no.
- **Registro de la prueba:** Deberá generarse una tabla con la información acerca de la prueba, los resultados y los posibles fallos encontrados. Esta tabla debe cumplir el formato definido en el apartado 7.4. Plantilla de informe de pruebas de este documento y ser incluida en el Informe de realización de pruebas del sistema. El documento con el resultado de las pruebas se adjuntará con la documentación del proyecto, pero no se incluirá en la memoria. El nombre de este documento será "Informe de Pruebas del Sistema.doc"





## **7.2.4 Necesidades del entorno**

Para que las pruebas definidas puedan ser ejecutadas de manera satisfactoria y produzcan los mismos resultados que en el entorno en el que el sistema real debe ser implantado, es necesario definir las características que debe tener el sistema donde se van a realizar las pruebas.

En el caso de este proyecto, el entorno de pruebas tendrá las mismas características que el entorno de implantación del sistema, por lo que en este apartado no es necesario definir nuevas necesidades para el entorno.

## **7.2.5 Criterio de aceptación/rechazo de un caso de prueba**

Al realizar la ejecución de cada uno de los casos de prueba definidos en el plan, es necesario determinar si la prueba se considera válida porque el sistema funciona como debería, o si presenta algún tipo de fallo en alguna de las funcionalidades requeridas.

El criterio establecido en este proyecto para considerar una prueba como *Aceptada*, considera necesario que dicha prueba se ejecute según lo definido, es decir, el sistema tendrá que responder de la manera exacta que detalle el caso de prueba correspondiente. En el caso de que el sistema produzca algún tipo de fallo o proporcione un resultado diferente al esperado, la prueba será *Rechazada*.



## 7.3 Definición de pruebas

### 7.3.1 Definición de pruebas de Aceptación del sistema

En este apartado se especifican las pruebas que se deben realizar al sistema para comprobar su correcto funcionamiento y que cumple con todas las especificaciones.

A continuación se describe el formato de tabla utilizado para la especificación de las pruebas y los distintos campos que contiene:

- **Identificador:** Nombra de manera única a la prueba de aceptación siguiendo el formato PR-A-XXX, donde XXX tomará valores numéricos dentro del rango 000-999.
- **Descripción:** Especifica el objetivo de la prueba.
- **Precondiciones:** Describe el estado en el que se debe encontrar el sistema de forma previa a la realización de la prueba.
- **Pasos a seguir:** Establece la secuencia de pasos que debe realizar el responsable de la prueba sobre el sistema.

Los procedimientos de prueba establecidos para este proyecto se describen a continuación.

Identificador	PR-A-001
Descripción	Arrancar la aplicación correctamente.
Precondiciones	Ninguna.
Pasos a seguir	1. Hacer doble clic sobre el icono de la aplicación. 2. Comprobar que la aplicación arranca correctamente.

Tabla 123. Prueba PR-A-001



Identificador	PR-A-002
Descripción	Seleccionar un fichero de entrada
Precondiciones	Aplicación arrancada. Disponer de un fichero de datos en formato Excel 2003
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Examinar..." situado junto al cuadro de texto "Fichero de entrada".</li><li>2. Seleccionar un fichero.</li><li>3. Hacer clic sobre el botón "Abrir".</li></ol> Comprobar que la ruta del fichero aparece correctamente en el cuadro de texto "Fichero de entrada".

Tabla 124. Prueba PR-A-002

Identificador	PR-A-003
Descripción	Seleccionar una ruta de salida
Precondiciones	Aplicación Arrancada.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Examinar..." situado junto al cuadro de texto "Ruta de salida".</li><li>2. Seleccionar una carpeta.</li><li>3. Hacer clic sobre "Aceptar".</li><li>4. Comprobar que la ruta de salida aparece correctamente en el cuadro de texto "Ruta de salida".</li></ol>

Tabla 125. Prueba PR-A-003

Identificador	PR-A-004
Descripción	Importar datos de un fichero de entrada con formato correcto (ver Anexo B: Formato para ficheros de entrada)
Precondiciones	Aplicación arrancada. Fichero de entrada seleccionado.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar la hoja en la que se encuentren los datos en el cuadro de texto "Hoja del libro Excel".</li><li>2. Hacer clic sobre el botón "Importar datos".</li><li>3. Comprobar que la aplicación muestra la ventana de selección de atributos.</li><li>4. Seleccionar al menos dos atributos entre los disponibles.</li><li>5. Hacer clic sobre el botón "Importar".</li><li>6. Comprobar que los atributos seleccionados aparecen correctamente en la lista "Atributos del fichero de datos".</li></ol>

Tabla 126. Prueba PR-A-004



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	PR-A-005
Descripción	Importar datos de un fichero de entrada con formato correcto (ver Anexo B: Formato para ficheros de entrada) sin seleccionar una hoja de datos.
Precondiciones	Aplicación arrancada con un fichero de entrada seleccionado.
Pasos a seguir	<ol style="list-style-type: none"><li>1. No seleccionar ninguna hoja en el cuadro de texto "Hoja del libro Excel".</li><li>2. Hacer clic sobre el botón "Importar datos".</li><li>3. Comprobar que la aplicación muestra el mensaje:</li><li>4. "Debe seleccionar una hoja del libro Excel para importar sus datos".</li></ol>

Tabla 127. Prueba PR-A-005

Identificador	PR-A-006
Descripción	Seleccionar un atributo de salida
Precondiciones	Aplicación arrancada. Al menos dos atributos importados a la lista "Atributos del fichero de datos".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar un atributo de la lista "Atributos del fichero de datos".</li><li>2. Hacer clic sobre el botón "Seleccionar salida(s)".</li><li>3. Comprobar que el atributo seleccionado aparece correctamente en la lista "Atributos de salida".</li></ol>

Tabla 128. Prueba PR-A-006

Identificador	PR-A-007
Descripción	Seleccionar múltiples atributos de salida.
Precondiciones	Aplicación arrancada. Al menos tres atributos importados a la lista "Atributos del fichero de datos".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar dos atributos de la lista "Atributos del fichero de datos".</li><li>2. Hacer clic sobre el botón "Seleccionar salida(s)".</li><li>3. Comprobar que los atributos seleccionados aparecen correctamente en la lista "Atributos de salida".</li></ol>

Tabla 129. Prueba PR-A-007



Identificador	PR-A-008
Descripción	Seleccionar un atributo de salida sin importar datos a la lista "Atributos del fichero de datos".
Precondiciones	Aplicación arrancada sin atributos importados a la lista "Atributos del fichero de datos".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Seleccionar salida(s)".</li><li>2. Comprobar que la aplicación muestra el siguiente mensaje: "Se debe cargar un fichero de datos y tener al menos un atributo de la lista seleccionado".</li></ol>

Tabla 130. Prueba PR-A-008

Identificador	PR-A-009
Descripción	Seleccionar un atributo de salida sin seleccionar ningún atributo de la lista "Atributos del fichero de datos".
Precondiciones	Aplicación arrancada sin seleccionar atributos de la lista "Atributos del fichero de datos".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Seleccionar salida(s)".</li><li>2. Comprobar que la aplicación muestra el siguiente mensaje: "Se debe cargar un fichero de datos y tener al menos un atributo de la lista seleccionado".</li></ol>

Tabla 131. Prueba PR-A-009

Identificador	PR-A-010
Descripción	Cargar clases del atributo para reparto de instancias.
Precondiciones	Aplicación arrancada con atributos importados a la lista "Atributos del fichero de datos".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar un atributo de la lista "Atributos del fichero de datos".</li><li>2. Hacer clic sobre el botón "Cargar clases para reparto".</li><li>3. Comprobar que el atributo seleccionado aparece correctamente en el cuadro de texto "Atributo seleccionado" y aparecen las clases y número de instancias existentes de cada tipo en la tabla de "Reparto de instancias".</li></ol>

Tabla 132. Prueba PR-A-010



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	PR-A-011
Descripción	Cargar clases del atributo para reparto de instancias sin seleccionar ningún atributo de la lista "Atributos del fichero de datos".
Precondiciones	Aplicación arrancada con atributos importados a la lista "Atributos del fichero de datos".
Pasos a seguir	<ol style="list-style-type: none"><li>1. No seleccionar ningún atributo de la lista "Atributos del fichero de datos".</li><li>2. Hacer clic sobre el botón "Cargar clases para reparto".</li><li>3. Comprobar que la aplicación muestra el siguiente mensaje: "Se debe cargar un fichero de datos y tener un atributo de la lista seleccionado".</li></ol>

Tabla 133. Prueba PR-A-011

Identificador	PR-A-012
Descripción	Cargar clases del atributo para reparto de instancias sin importar ningún atributo a la lista "Atributos del fichero de datos".
Precondiciones	Aplicación arrancada sin atributos importados a la lista "Atributos del fichero de datos".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Cargar clases para reparto".</li><li>2. Comprobar que la aplicación muestra el siguiente mensaje: "Se debe cargar un fichero de datos y tener un atributo de la lista seleccionado".</li></ol>

Tabla 134. Prueba PR-A-012

Identificador	PR-A-013
Descripción	Asignar los porcentajes de reparto de instancias de forma aleatoria.
Precondiciones	Aplicación arrancada con un atributo seleccionado para el reparto de instancias (ver PR-A-010).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Porcentajes aleatorios".</li><li>2. Comprobar que a cada clase se le asigna un porcentaje mayor que cero, y que la suma de todos los porcentajes sea igual a cien.</li></ol>

Tabla 135. Prueba PR-A-013



Identificador	PR-A-014
Descripción	Repartir las instancias disponibles entre los conjuntos a generar.
Precondiciones	Aplicación arrancada con los datos del fichero de entrada importados.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Introducir un número de instancias mayor que cero para los conjuntos de entrenamiento, validación y test.</li><li>2. Comprobar que los valores se pueden introducir correctamente y que la etiqueta de texto "Restantes" va decrementando el número de instancias restantes.</li></ol>

Tabla 136. Prueba PR-A-014

Identificador	PR-A-015
Descripción	Añadir un experimento a la lista "Experimentos a generar".
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>2. Comprobar que el experimento configurado se inserta correctamente en la última posición de la lista "Experimentos a generar" con un nombre que incluya la siguiente información:<ul style="list-style-type: none"><li>• Número de experimento.</li><li>• Número de pruebas.</li><li>• Técnica de división.</li><li>• Número de instancias de los conjuntos.</li><li>• Atributo clase.</li><li>• Porcentajes de reparto de las instancias.</li></ul></li></ol>

Tabla 137. Prueba PR-A-015



Identificador	PR-A-016
Descripción	Añadir un experimento a la lista "Experimentos a generar" sin seleccionar atributos de salida.
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento excepto los atributos de salida.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>2. Comprobar que el experimento no se añade a la lista "Experimentos a generar" y la aplicación muestra el siguiente mensaje: "Debe cargarse un fichero de entrada, seleccionar al menos un atributo de salida, y establecer el reparto de clases".</li></ol>

Tabla 138. Prueba PR-A-016

Identificador	PR-A-017
Descripción	Añadir un experimento a la lista "Experimentos a generar" sin seleccionar un fichero de entrada.
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento excepto el fichero de entrada.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>2. Comprobar que el experimento no se añade a la lista "Experimentos a generar" y la aplicación muestra el siguiente mensaje: "Debe cargarse un fichero de entrada, seleccionar al menos un atributo de salida, y establecer el reparto de clases".</li></ol>

Tabla 139. Prueba PR-A-017

Identificador	PR-A-018
Descripción	Añadir un experimento a la lista "Experimentos a generar" sin realizar el reparto de instancias.
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento excepto el reparto de instancias.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>2. Comprobar que el experimento no se añade a la lista "Experimentos a generar" y la aplicación muestra el siguiente mensaje: "Debe cargarse un fichero de entrada, seleccionar al menos un atributo de salida, y establecer el reparto de clases".</li></ol>

Tabla 140. Prueba PR-A-018





Identificador	PR-A-019
Descripción	Añadir un experimento a la lista "Experimentos a generar" con un valor no numérico en el campo "Número de pruebas".
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento.
Pasos a seguir	<ol style="list-style-type: none"><li>1. En el campo "Número de pruebas" introducir un valor no numérico: "abc".</li><li>2. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>3. Comprobar que el experimento no se añade a la lista "Experimentos a generar" y la aplicación muestra el siguiente mensaje: "Los números de pruebas e instancias deben contener valores numéricos".</li></ol>

Tabla 141. Prueba PR-A-019

Identificador	PR-A-020
Descripción	Añadir un experimento a la lista "Experimentos a generar" con un valor en el campo "Número de pruebas" menor o igual a cero.
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento.
Pasos a seguir	<ol style="list-style-type: none"><li>1. En el campo "Número de pruebas" introducir el valor: 0.</li><li>2. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>3. Comprobar que el experimento no se añade a la lista "Experimentos a generar" y la aplicación muestra el siguiente mensaje: "No se puede añadir un experimento con número de pruebas menor o igual a cero"</li></ol>

Tabla 142. Prueba PR-A-020



Identificador	PR-A-021
Descripción	Añadir un experimento a la lista "Experimentos a generar" con una cantidad de instancias repartidas superior a la de instancias disponibles.
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Introducir valores en los conjuntos de entrenamiento, validación y test, cuya suma total sea superior al valor de "Número de instancias".</li><li>2. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>3. Comprobar que el experimento no se añade a la lista "Experimentos a generar" y la aplicación muestra el siguiente mensaje: "La suma del número de instancias repartidas es superior al número de instancias disponibles".</li></ol>

Tabla 143. Prueba PR-A-021

Identificador	PR-A-022
Descripción	Añadir un experimento a la lista "Experimentos a generar" con porcentajes de reparto de instancias cuya suma no sea 100.
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Introducir valores para los porcentajes de reparto de clases cuya suma sea distinta de 100.</li><li>2. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>3. Comprobar que el experimento no se añade a la lista "Experimentos a generar" y la aplicación muestra el siguiente mensaje: "El reparto de porcentajes entre las clases debe sumar 100%".</li></ol>

Tabla 144. Prueba PR-A-022



Identificador	PR-A-023
Descripción	Añadir varios experimentos a la lista "Experimentos a generar" de forma automática.
Precondiciones	Aplicación arrancada, habiendo configurado todos los parámetros de configuración de un experimento.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Introducir un valor superior a cero en el cuadro de texto "Número de decrementos" y en al menos uno de los decrementos asociados a los conjuntos de entrenamiento, validación y test.</li><li>2. Hacer clic sobre el botón "Añadir experimento(s)".</li><li>3. Comprobar que los experimentos configurado se insertan al final de la lista "Experimentos a generar" con un nombre que incluya la siguiente información:<ul style="list-style-type: none"><li>• Número de experimento.</li><li>• Número de pruebas.</li><li>• Técnica de división.</li><li>• Número de instancias de los conjuntos.</li><li>• Atributo de reparto.</li><li>• Porcentajes de reparto de las instancias.</li></ul></li><li>4. Comprobar que los experimentos añadidos cumplen con los decrementos asignados en el paso 1.</li></ol>

Tabla 145. Prueba PR-A-023

Identificador	PR-A-024
Descripción	Eliminar un experimento de la lista "Experimentos a generar".
Precondiciones	Aplicación arrancada, habiendo añadido al menos un experimento a la lista "Experimentos a generar".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar el experimento a eliminar.</li><li>2. Hacer clic sobre el botón "Eliminar experimento(s)".</li><li>3. Comprobar que la aplicación muestra el siguiente mensaje: "Experimento(s) eliminado(s) correctamente" y que el experimento seleccionado desaparece de la lista "Experimentos a generar".</li></ol>

Tabla 146. Prueba PR-A-024



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	PR-A-025
Descripción	Eliminar varios experimentos de la lista "Experimentos a generar".
Precondiciones	Aplicación arrancada, habiendo añadido al menos dos experimentos a la lista "Experimentos a generar".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar al menos dos experimentos a eliminar.</li><li>2. Hacer clic sobre el botón "Eliminar experimento(s)".</li><li>3. Comprobar que la aplicación muestra el siguiente mensaje: "Experimento(s) eliminado(s) correctamente" y que los experimentos seleccionados desaparecen de la lista "Experimentos a generar".</li></ol>

Tabla 147. Prueba PR-A-025

Identificador	PR-A-026
Descripción	Eliminar un experimento de la lista "Experimentos a generar" sin seleccionar ningún experimento
Precondiciones	Aplicación arrancada, habiendo añadido al menos un experimento a la lista "Experimentos a generar".
Pasos a seguir	<ol style="list-style-type: none"><li>1. No seleccionar ningún experimento de la lista "Experimentos a generar".</li><li>2. Hacer clic sobre el botón "Eliminar experimento(s)".</li><li>3. Comprobar que la aplicación muestra el siguiente mensaje: "No hay experimentos seleccionados para eliminar".</li></ol>

Tabla 148. Prueba PR-A-026

Identificador	PR-A-027
Descripción	Guardar configuración de la aplicación.
Precondiciones	Aplicación arrancada, habiendo añadido al menos un experimento a la lista "Experimentos a generar", y habiendo seleccionado tanto fichero de entrada como ruta de salida.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Guardar configuración".</li><li>2. Introducir un nombre de archivo o seleccionar uno existente.</li><li>3. Hacer clic sobre el botón "Guardar".</li><li>4. Comprobar que la aplicación muestra el siguiente mensaje: "Fichero de configuración guardado correctamente".</li></ol>

Tabla 149. Prueba PR-A-027



Identificador	PR-A-028
Descripción	Guardar configuración de la aplicación sin seleccionar un fichero de entrada.
Precondiciones	Aplicación arrancada, habiendo seleccionado una ruta de salida.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Guardar configuración".</li><li>2. Comprobar que la aplicación muestra el siguiente mensaje: "Debe tener al menos la localización de los ficheros y un experimento para guardar la configuración".</li></ol>

Tabla 150. Prueba PR-A-028

Identificador	PR-A-029
Descripción	Guardar configuración de la aplicación sin seleccionar una ruta de salida.
Precondiciones	Aplicación arrancada, habiendo seleccionado un fichero de entrada y habiendo añadido al menos un experimento a la lista "Experimentos a generar".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Guardar configuración".</li><li>2. Comprobar que la aplicación muestra el siguiente mensaje: "Debe tener al menos la localización de los ficheros y un experimento para guardar la configuración".</li></ol>

Tabla 151. Prueba PR-A-029

Identificador	PR-A-030
Descripción	Guardar configuración de la aplicación sin añadir ningún experimento a la lista "Experimentos a generar".
Precondiciones	Aplicación arrancada, habiendo seleccionado un fichero de entrada y una ruta de salida.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Guardar configuración".</li><li>2. Comprobar que la aplicación muestra el siguiente mensaje: "Debe tener al menos la localización de los ficheros y un experimento para guardar la configuración".</li></ol>

Tabla 152. Prueba PR-A-030



Identificador	PR-A-031
Descripción	Cargar configuración de la aplicación.
Precondiciones	Aplicación arrancada, habiendo almacenado previamente la configuración en un fichero (ver PR-A-027).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Cargar configuración".</li><li>2. Seleccionar el fichero de configuración existente.</li><li>3. Hacer clic sobre el botón "Abrir".</li><li>4. Comprobar que la aplicación carga la configuración correctamente y muestra el siguiente mensaje: "Fichero de configuración cargado correctamente".</li></ol>

Tabla 153. Prueba PR-A-031

Identificador	PR-A-032
Descripción	Generar un único experimento de la lista "Experimentos a generar".
Precondiciones	Aplicación arrancada, habiendo añadido al menos dos experimentos a la lista "Experimentos a generar".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar un experimento de la lista "Experimentos a generar".</li><li>2. Hacer clic sobre el botón "Generar Seleccionados".</li><li>3. Comprobar que la aplicación muestra el mensaje: "Ficheros generados correctamente".</li><li>4. Comprobar en la ruta de salida introducida que únicamente se han generado correctamente los archivos correspondientes al experimento seleccionado.</li></ol>

Tabla 154. Prueba PR-A-032



Identificador	PR-A-033
Descripción	Generar todos los experimentos de la lista "Experimentos a generar".
Precondiciones	Aplicación arrancada, habiendo añadido al menos dos experimentos a la lista "Experimentos a generar".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Seleccionar Todos".</li><li>2. Comprobar que se han seleccionado todos los experimentos de la lista "experimentos a generar".</li><li>3. Hacer clic sobre el botón "Generar Seleccionados".</li><li>4. Comprobar que la aplicación muestra el mensaje: "Ficheros generados correctamente".</li><li>5. Comprobar en la ruta de salida introducida que se han generado correctamente los archivos para cada uno de los experimentos de la lista "Experimentos a generar".</li></ol>

Tabla 155. Prueba PR-A-033

Identificador	PR-A-034
Descripción	Generar los experimentos seleccionados de la lista "Experimentos a generar" sin ningún experimento añadido.
Precondiciones	Aplicación arrancada, no habiendo añadido ningún experimento a la lista "Experimentos a generar".
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Generar Seleccionados".</li><li>2. Comprobar que la aplicación muestra el mensaje: "Debe añadir al menos un experimento para generar los ficheros".</li></ol>

Tabla 156. Prueba PR-A-034

Identificador	PR-A-035
Descripción	Salir de la aplicación.
Precondiciones	Aplicación arrancada.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Salir".</li><li>2. Comprobar que la aplicación se cierra correctamente.</li></ol>

Tabla 157. Prueba PR-A-035



Identificador	PR-A-036
Descripción	Cargar los ficheros de red.
Precondiciones	Aplicación arrancada, habiendo generado previamente al menos un experimento (ver PR-A-032).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Examinar..." situado junto al cuadro de texto "Fichero de Red:".</li><li>2. Seleccionar la carpeta que contiene los ficheros de red.</li><li>3. hacer clic sobre "Aceptar".</li><li>4. Comprobar que la ruta de salida aparece correctamente en el cuadro de texto "Fichero de Red:".</li><li>5. Comprobar que en la lista "Tipología Red" aparecen los ficheros de red junto a su tipología de red.</li></ol>

Tabla 158. Prueba PR-A-036

Identificador	PR-A-037
Descripción	Cambiar tipología de red seleccionado un fichero de red.
Precondiciones	Aplicación arrancada, habiendo identificado al menos un fichero de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar un fichero de red de la lista "Tipología Red".</li><li>2. Hacer clic sobre el botón "Cambiar Tipología Red".</li><li>3. Seleccionar una tipología de red y hacer clic en aceptar.</li><li>4. Comprobar que el fichero de red seleccionado ha cambiado a la tipología seleccionada.</li></ol>

Tabla 159. Prueba PR-A-037

Identificador	PR-A-038
Descripción	Cambiar tipología de red no seleccionando ningún fichero de red.
Precondiciones	Aplicación arrancada, habiendo identificado al menos un fichero de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Cambiar Tipología Red".</li><li>2. Comprobar que la aplicación muestra el mensaje "Debe seleccionar un fichero de red".</li></ol>

Tabla 160. Prueba PR-A-038





Identificador	PR-A-039
Descripción	Cambiar tipología de red seleccionando dos ficheros de red.
Precondiciones	Aplicación arrancada, habiendo identificado al menos dos ficheros de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar dos ficheros de red de la lista "Tipología Red".</li><li>2. Hacer clic sobre el botón "Cambiar Tipología Red".</li><li>3. Comprobar que la aplicación muestra el mensaje "Debe seleccionar un fichero de red".</li></ol>

Tabla 161. Prueba PR-A-039

Identificador	PR-A-040
Descripción	Seleccionar error cuadrático medio como criterio de parada de entrenamiento en experimentos sin validación cruzada.
Precondiciones	Aplicación arrancada, habiendo generado previamente al menos un experimento (ver PR-A-032) que no tenga validación cruzada.
Pasos a seguir	<ol style="list-style-type: none"><li>1. Comprobar que no se puede seleccionar "Cross Validation MSE" como criterio de parada.</li><li>2. Cambiar el valor para el campo "MSE".</li><li>3. Cambiar el valor de el campo "Nº de ciclos máx.:"</li><li>4. Comprobar que los valores aparecen correctamente.</li></ol>

Tabla 162. Prueba PR-A-040

Identificador	PR-A-041
Descripción	Seleccionar error cuadrático medio como criterio de parada de entrenamiento en experimentos con validación cruzada.
Precondiciones	Aplicación arrancada, habiendo generado previamente al menos un experimento (ver PR-A-032) que tenga validación cruzada
Pasos a seguir	<ol style="list-style-type: none"><li>1. Comprobar que se puede seleccionar "Cross Validation MSE" como criterio de parada.</li><li>2. Cambiar el valor para el campo "MSE".</li><li>3. Cambiar el valor de el campo "Nº de ciclos máx.:"</li><li>4. Comprobar que los valores aparecen correctamente.</li></ol>

Tabla 163. Prueba PR-A-041



Identificador	PR-A-042
Descripción	Seleccionar error cuadrático medio en base a la validación cruzada como criterio de parada de entrenamiento.
Precondiciones	Aplicación arrancada, habiendo generado previamente al menos un experimento (ver PR-A-032) que tenga validación cruzada
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar "Cross Validation MSE" como criterio de parada.</li><li>2. Cambiar el valor de el campo "Nº de ciclos máx.:"</li><li>3. Comprobar que los valores aparecen correctamente.</li></ol>

Tabla 164. Prueba PR-A-042

Identificador	PR-A-043
Descripción	Entrenar un experimento seleccionado en una red seleccionada.
Precondiciones	Aplicación arrancada, habiendo identificado al menos un fichero de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar un fichero de red de la lista "Tipología Red".</li><li>2. Seleccionar un experimento de la lista "Experimentos Disponibles:".</li><li>3. Deseleccionar la opción "crear Ficheros de Resumen y Evaluación".</li><li>4. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>5. Comprobar que la aplicación muestra la interfaz con la información del entrenamiento.</li><li>6. Una vez que se vuelve a mostrar la interfaz de entrenamiento comprobar que se han generado los ficheros de salida correspondientes correctamente.</li></ol>

Tabla 165. Prueba PR-A-043



Identificador	PR-A-044
Descripción	Entrenar un experimento seleccionado en dos redes seleccionadas.
Precondiciones	Aplicación arrancada, habiendo identificado al menos dos ficheros de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar dos ficheros de red de la lista "Tipología Red".</li><li>2. Seleccionar un experimento de la lista "Experimentos Disponibles:".</li><li>3. Deseleccionar la opción "crear Ficheros de Resumen y Evaluación".</li><li>4. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>5. Comprobar que la aplicación muestra la interfaz con la información del entrenamiento.</li><li>6. Una vez que se vuelve a mostrar la interfaz de entrenamiento comprobar que se han generado los ficheros de salida correspondientes correctamente.</li></ol>

Tabla 166. Prueba PR-A-044

Identificador	PR-A-045
Descripción	Entrenar dos experimentos en una red seleccionada.
Precondiciones	Aplicación arrancada, habiendo generado al menos dos experimentos ver (PR-A-033) e identificado al menos un fichero de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar dos ficheros de red de la lista "Tipología Red".</li><li>2. Seleccionar dos experimentos de la lista "Experimentos Disponibles:".</li><li>3. Deseleccionar la opción "crear Ficheros de Resumen y Evaluación".</li><li>4. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>5. Comprobar que la aplicación muestra la interfaz con la información del entrenamiento.</li><li>6. Una vez que se vuelve a mostrar la interfaz de entrenamiento comprobar que se han generado los ficheros de salida correspondientes correctamente.</li></ol>

Tabla 167. Prueba PR-A-045



Identificador	PR-A-046
Descripción	Entrenar dos experimentos en dos redes seleccionadas.
Precondiciones	Aplicación arrancada, habiendo generado al menos dos experimentos ver (PR-A-033) e identificado al menos dos ficheros de red (ver PR-A-038).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar dos ficheros de red de la lista "Tipología Red".</li><li>2. Seleccionar dos experimentos de la lista "Experimentos Disponibles:".</li><li>3. Deseleccionar la opción "crear Ficheros de Resumen y Evaluación".</li><li>4. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>5. Comprobar que la aplicación muestra la interfaz con la información del entrenamiento.</li><li>6. Una vez que se vuelve a mostrar la interfaz de entrenamiento comprobar que se han generado los ficheros de salida correspondientes correctamente.</li></ol>

Tabla 168. Prueba PR-A-046

Identificador	PR-A-047
Descripción	Seleccionar una ruta para el fichero de resumen.
Precondiciones	Aplicación arrancada, habiendo generado previamente al menos un experimento (ver PR-A-034).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón "Examinar..." situado junto al cuadro de texto "Ruta Fichero Resumen".</li><li>2. Seleccionar una carpeta.</li><li>3. Hacer clic sobre el botón "Aceptar".</li></ol> Comprobar que la ruta del fichero aparece correctamente en el cuadro de texto "Ruta Fichero Resumen".

Tabla 169. Prueba PR-A-047



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	PR-A-048
Descripción	Entrenar un experimento seleccionado en una red seleccionada realizando los informes de evaluación y resumen.
Precondiciones	Aplicación arrancada, habiendo identificado al menos un fichero de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar un fichero de red de la lista "Tipología Red".</li><li>2. Seleccionar un experimento de la lista "Experimentos Disponibles:".</li><li>3. Comprobar que esta seleccionada la opción "crear Ficheros de Resumen y Evaluación".</li><li>4. Comprobar que la ruta del fichero de resumen es correcta, o cambiar la ruta del fichero de resumen (ver PR-A-048).</li><li>5. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>6. Comprobar que la aplicación muestra la interfaz con la información del entrenamiento.</li><li>7. Comprobar que la interfaz muestra el mensaje "Generando los Ficheros de evaluación y resumen".</li><li>8. Una vez que se vuelve a mostrar la interfaz de entrenamiento comprobar que se han generado los ficheros de salida correspondientes correctamente y que han generado los informes de evaluación y resumen.</li></ol>

Tabla 170. Prueba PR-A-048



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	PR-A-049
Descripción	Entrenar un experimento seleccionado en dos redes seleccionadas realizando los informes de evaluación y resumen.
Precondiciones	Aplicación arrancada, habiendo identificado al menos dos ficheros de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar dos ficheros de red de la lista "Tipología Red".</li><li>2. Seleccionar un experimento de la lista "Experimentos Disponibles:".</li><li>3. Comprobar que esta seleccionada la opción "crear Ficheros de Resumen y Evaluación".</li><li>4. Comprobar que la ruta del fichero de resumen es correcta, o cambiar la ruta del fichero de resumen (ver PR-A-048).</li><li>5. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>6. Comprobar que la aplicación muestra la interfaz con la información del entrenamiento.</li><li>7. Comprobar que la interfaz muestra el mensaje "Generando los Ficheros de evaluación y resumen".</li><li>8. Una vez que se vuelve a mostrar la interfaz de entrenamiento comprobar que se han generado los ficheros de salida correspondientes correctamente y que han generado los informes de evaluación y resumen.</li></ol>

Tabla 171. Prueba PR-A-049



Identificador	PR-A-050
Descripción	Entrenar dos experimentos en una red seleccionada realizando los informes de evaluación y resumen.
Precondiciones	Aplicación arrancada, habiendo generado al menos dos experimentos ver (PR-A-033) e identificado al menos un fichero de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar dos ficheros de red de la lista "Tipología Red".</li><li>2. Seleccionar dos experimentos de la lista "Experimentos Disponibles:".</li><li>3. Comprobar que esta seleccionada la opción "crear Ficheros de Resumen y Evaluación".</li><li>4. Comprobar que la ruta del fichero de resumen es correcta, o cambiar la ruta del fichero de resumen (ver PR-A-048).</li><li>5. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>6. Comprobar que la aplicación muestra la interfaz con la información del entrenamiento.</li><li>7. Comprobar que la interfaz muestra el mensaje "Generando los Ficheros de evaluación y resumen".</li><li>8. Una vez que se vuelve a mostrar la interfaz de entrenamiento comprobar que se han generado los ficheros de salida correspondientes correctamente y que han generado los informes de evaluación y resumen.</li></ol>

Tabla 172. Prueba PR-A-050



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	PR-A-051
Descripción	Entrenar dos experimentos en dos redes seleccionadas realizando los informes de evaluación y resumen.
Precondiciones	Aplicación arrancada, habiendo generado al menos dos experimentos ver (PR-A-033) e identificado al menos dos ficheros de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar dos ficheros de red de la lista "Tipología Red".</li><li>2. Seleccionar dos experimentos de la lista "Experimentos Disponibles:".</li><li>3. Comprobar que esta seleccionada la opción "crear Ficheros de Resumen y Evaluación".</li><li>4. Comprobar que la ruta del fichero de resumen es correcta, o cambiar la ruta del fichero de resumen (ver PR-A-048).</li><li>5. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>6. Comprobar que la aplicación muestra la interfaz con la información del entrenamiento.</li><li>7. Comprobar que la interfaz muestra el mensaje "Generando los Ficheros de evaluación y resumen".</li><li>8. Una vez que se vuelve a mostrar la interfaz de entrenamiento comprobar que se han generado los ficheros de salida correspondientes correctamente y que han generado los informes de evaluación y resumen.</li></ol>

Tabla 173. Prueba PR-A-051

Identificador	PR-A-052
Descripción	Entrenar un experimento sin seleccionar ninguna red.
Precondiciones	Aplicación arrancada, habiendo generado previamente al menos un experimento (ver PR-A-032).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar un experimento de la lista "Experimentos Disponibles:".</li><li>2. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>3. Comprobar que la aplicación muestra el mensaje. "Debe seleccionar al menos una tipología de red".</li></ol>

Tabla 174. Prueba PR-A-052





## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Identificador	PR-A-053
Descripción	Entrenar sin seleccionar ningún experimento en una red seleccionada.
Precondiciones	Aplicación arrancada, habiendo identificado al menos un fichero de red (ver PR-A-036).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar un fichero de red de la lista "Tipología Red".</li><li>2. Hacer clic sobre el botón "Entrenar Seleccionados".</li><li>3. Comprobar que la aplicación muestra el mensaje: "Debe seleccionar al menos una tipología de red".</li></ol>

Tabla 175. Prueba PR-A-053

Identificador	PR-A-054
Descripción	Abrir fichero evaluación sin seleccionar ningún experimento.
Precondiciones	Aplicación arrancada, habiendo entrenado al menos un experimento en una red seleccionada (ver PR-A-048).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Deseleccionar los experimentos de la lista "Experimentos Disponibles:".</li><li>2. Hacer clic sobre el botón Abrir fichero Evaluación.</li><li>3. Comprobar que la aplicación muestra el mensaje: "Seleccione un experimento para ver su evaluación".</li></ol>

Tabla 176. Prueba PR-A-054

Identificador	PR-A-055
Descripción	Abrir fichero evaluación seleccionado al menos dos experimentos
Precondiciones	Aplicación arrancada, habiendo entrenado al menos dos experimentos en una red seleccionada (ver PR-A-050).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar al menos dos experimentos de la lista "Experimentos Disponibles:".</li><li>2. Hacer clic sobre el botón Abrir fichero Evaluación.</li><li>3. Comprobar que la aplicación muestra el mensaje: "Seleccione un experimento para ver su evaluación".</li></ol>

Tabla 177. Prueba PR-A-055



Identificador	PR-A-056
Descripción	Abrir fichero evaluación seleccionado un experimento
Precondiciones	Aplicación arrancada, habiendo entrenado al menos un experimento en una red seleccionada (ver PR-A-048).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Seleccionar al menos un experimento de la lista "Experimentos Disponibles: ".</li><li>2. Hacer clic sobre el botón Abrir fichero Evaluación.</li><li>3. Comprobar que la aplicación el fichero solicitado.</li></ol>

Tabla 178. Prueba PR-A-056

Identificador	PR-A-057
Descripción	Abrir fichero resumen
Precondiciones	Aplicación arrancada, habiendo entrenado al menos un experimento en una red seleccionada (ver PR-A-048).
Pasos a seguir	<ol style="list-style-type: none"><li>1. Hacer clic sobre el botón Abrir fichero Resumen.</li><li>2. Comprobar que la aplicación el fichero solicitado.</li></ol>

Tabla 179. Prueba PR-A-057



### 7.3.1.1 Matriz de trazabilidad Pruebas Aceptación/Requisitos de Capacidad

	RU-C-001	RU-C-002	RU-C-003	RU-C-004	RU-C-005	RU-C-006	RU-C-007	RU-C-008	RU-C-009	RU-C-010	RU-C-011	RU-C-012	RU-C-013	RU-C-014	RU-C-015	RU-C-016	RU-C-017
PR-A-001	X																
PR-A-002		X															
PR-A-003		X															
PR-A-004			X														
PR-A-005			X														
PR-A-006				X													
PR-A-007				X													
PR-A-008				X													
PR-A-009				X													
PR-A-010				X													
PR-A-011				X													
PR-A-012				X													
PR-A-013				X													
PR-A-014				X													
PR-A-015					X	X											
PR-A-016						X											
PR-A-017						X											
PR-A-018						X											
PR-A-019						X											
PR-A-020						X											
PR-A-021						X											
PR-A-022						X											
PR-A-023							X										
PR-A-024								X									
PR-A-025								X									
PR-A-026								X									
PR-A-027									X								
PR-A-028									X								



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

PR-A-029									X								
PR-A-030									X								
PR-A-031									X								
PR-A-032										X							
PR-A-033										X							
PR-A-034										X							
PR-A-035											X						
PR-A-036												X					
PR-A-037													X				
PR-A-038													X				
PR-A-039													X				
PR-A-040														X			
PR-A-041														X			
PR-A-042														X			
PR-A-043															X		
PR-A-044															X		
PR-A-045															X		
PR-A-046															X		
PR-A-047																	X
PR-A-048															X	X	X
PR-A-049															X	X	X
PR-A-050															X	X	X
PR-A-051															X	X	X
PR-A-052															X		
PR-A-053															X		
PR-A-054																X	
PR-A-055																X	
PR-A-056																X	
PR-A-057																	X

Tabla 180. Matriz de trazabilidad Pruebas Aceptación – Requisitos de capacidad



### 7.3.2 Definición de las Pruebas Unitarias

Dado que las pruebas unitarias están muy ligadas a la codificación y debido a su amplitud no se detallarán en la documentación.

### 7.3.3 Definición de las pruebas de implantación

Las pruebas para comprobar que se ha implantado correctamente el sistema se centran en aquellas que son vitales para el usuario, es por ello que las pruebas que serán necesarias pasar obligatoriamente para comprobar el correcto funcionamiento del aplicativo un vez implantado serán *las pruebas de Aceptación*.

### 7.3.4 Definición de las pruebas de Integración

A continuación se describe el formato de tabla utilizado para la especificación de las pruebas y los distintos campos que contiene:

- **Identificador:** Nombra de manera única a la prueba de aceptación siguiendo el formato PR-I-XXX, donde XXX tomará valores numéricos dentro del rango 000-999.
- **Descripción:** Especifica el objetivo de la prueba.
- **Precondiciones:** Describe el estado en el que se debe encontrar el sistema de forma previa a la realización de la prueba.
- **Pasos a seguir:** Establece la secuencia de pasos que debe realizar el responsable de la prueba sobre el sistema.

Los procedimientos de prueba establecidos para este proyecto se describen a continuación.

Identificador	PR-A-001
Descripción	Aplicación Excel y su vinculación con la aplicación.
Precondiciones	
Pasos a seguir	Se comprueba que la aplicación interacciona correctamente al leer o escribir en documentos en formato Excel.

Tabla 181. Prueba PR-I-001



Identificador	PR-A-002
Descripción	Aplicación Neurosolutions y su vinculación con la aplicación
Precondiciones	
Pasos a seguir	Se comprueba que la aplicación se comunica correctamente con la aplicación Neurosolutions

Tabla 182. Prueba PR-I-002

### 7.3.5 Definición de las pruebas de Sistema

En este apartado definiremos las pruebas que se deben realizar para comprobar el correcto funcionamiento de los componentes del sistema.

Para la realización de estas pruebas se han utilizado dos dominios de aplicación distintos: el impacto balístico y wine quality [10] (calidad del vino). En los siguientes apartados se detallan en más profundidad tanto los dominios utilizados como el funcionamiento del sistema.

Como se podrá observar, las pruebas se han dividido en Clasificación y Regresión, los dos tipos de predicción que permite el sistema

#### 7.3.5.1 Prueba de dominio A: Clasificación

Para la prueba de Clasificación se ha utilizado el dominio del wine quality (calidad del vino). En concreto los datos tratan sobre el vino tinto portugués "Vinho Verde". Se dispondrán de 1599 instancias. En cuanto a los parámetros de entrada contamos con los siguientes:

- "fixed acidity" (acidez fija).
- "volatile acidity" (acidez volátil).
- "citric acid" (ácido cítrico).
- "residual sugar" (azúcar residual).
- "chlorides" (cloruros).
- "free sulfur dioxide" (dióxido de azufre libre).
- "total sulfur dioxide" (dióxido de azufre total).



- “density” (densidad).
- “pH”.
- “sulphates” (sulfatos).
- “alcohol”.

El propósito de la prueba es determinar, dados los parámetros de entrada la calidad del vino. La variable a predecir por el sistema es “quality” (calidad) teniendo una puntuación entre 1 y 10.

A continuación definiremos los parámetros de la prueba:

**Características del entrenamiento:**

- Red neuronal: Multilayer Perceptron, Jordam-Elman, Radial basis function, Self-organizing feature map, Recurrent Network, Support Vector Machine.
- Datos de entrada: “fixed acidity”, “volatile acidity”, “citric acid”, “residual sugar”, “chlorides”, “free sulfur dioxide”, “total sulfur dioxide”, “density”, “pH”, “sulphates”, “alcohol”.
- Datos de salida: “quality”.
- Porcentaje de ensayos “quality”: totalmente aleatorio.
- N° epochs: 1000.

**Plan de pruebas A:**

N° de pruebas	N° ensayos Train	N° ensayos Test	N° ensayos de validación cruzada
200	1000	400	150
150	1000	400	150
100	1000	400	150
50	1000	400	150

Tabla 183. Plan de pruebas A de dominio A



**Plan de pruebas B:**

Nº de pruebas	Nº ensayos Train	Nº ensayos Test	Nº ensayos de validación cruzada
200	800	600	150
150	800	600	150
100	800	600	150
50	800	600	150

Tabla 184. Plan de pruebas B de dominio A

### 7.3.5.2 Prueba de dominio B: Regresión

Para la prueba de Regresión el dominio utilizado ha sido del impacto balístico. Disponemos de 220 instancias. Como variables tenemos:

- “Nº Ensayo”: Es el número de ensayo (de 1 a 220).
- “Grosor Placa (mm)”: Es el grosor de la placa expresado en milímetros.
- “Radio Proy (mm)”: Es el radio del proyectil expresado en milímetros.
- “Long Proy (mm)”: Es la longitud del proyectil expresada en milímetros.
- “Vel (m/s)”: Es la velocidad del proyectil expresada en metros por segundo.
- “Masa Inicial (g)”: Es la masa del proyectil antes del impacto expresada en gramos.
- “Material Proy”: Es el material del proyectil, codificado mediante un número.
- “Material Placa”: Es el material de la placa, codificado mediante un número, al igual que en el caso del material del proyectil.
- “Masa Final (g)”: Es la masa del proyectil una vez que ha impactado sobre la placa expresada en gramos.





- “(Mi-Mf)/Mi (%)”: Es el porcentaje de pérdida de masa del proyectil tras impactar sobre la placa.
- “Penetración”: Es un campo booleano que indica si el proyectil ha atravesado completamente la placa (perforación), indicándolo mediante un “si” en ese caso, y un “no” en caso contrario (penetración).
- “Vres (m/s)”: Es la velocidad residual del proyectil cuando sí atraviesa la placa completamente, expresada en metros por segundo.
- “DOP (mm)”: Representa la profundidad de penetración en la placa expresada en milímetros, en el caso de que el proyectil no la atravesase completamente.

El objetivo de este plan es comprobar el grado de ajuste que se puede conseguir de las variables “(Mi-Mf)/Mi (%)” y “Vres (m/s)” modificando alguna característica de su entrenamiento. De este modo, se han realizado una serie de experimentos modificando solamente el valor del número de ensayos de entrenamiento manteniendo el resto de la configuración sin cambios.

#### **Características del entrenamiento:**

- Red neuronal: Multilayer Perceptron, Jordam-Elman, Radial basis function, Self-organizing feature map, Recurrent Network.
- Datos de entrada: “Grosor Placa (mm)”, “Radio Proy (mm)”, “Long Proy (mm)”, “Vel (m/s)”, “Material Proy”, “Material Placa”
- Datos de salida: “(Mi-Mf)/Mi (%)” y “Vres (m/s)”
- N° epochs: 1000



---

---

**Plan de pruebas A:**

Nº de pruebas	Nº ensayos Train	Nº ensayos Test	Nº ensayos de validación cruzada
300	110	35	15
250	110	35	15
200	110	35	15
150	110	35	15
100	110	35	15
50	110	35	15

Tabla 185. Plan de pruebas A de dominio B

**Plan de pruebas B:**

Nº de pruebas	Nº ensayos Train	Nº ensayos Test	Nº ensayos de validación cruzada
300	80	50	30
250	80	50	30
200	80	50	30
150	80	50	30
100	80	50	30
50	80	50	30

Tabla 186. Plan de pruebas B de dominio B



## 7.4 *Plantilla de informe de pruebas*

Durante la realización de las pruebas se debe realizar un informe en el que se detallen los resultados obtenidos. A continuación se especifican los campos que deben ser incluidos en el informe y el formato establecido para este proyecto:

- **Identificador:** Identificador de la prueba realizada.
- **Fecha:** Fecha en la que se realizó la prueba.
- **Responsable:** Miembro del equipo del proyecto encargado de realizar la prueba.
- **Descripción:** Objetivo de la prueba realizada.
- **Resultado:** Este campo mostrará el valor Aceptada si el sistema ha realizado la prueba correctamente, o Rechazada en caso contrario.
- **Descripción de los fallos:** Si la prueba ha sido Rechazada este campo contendrá una descripción de los fallos encontrados durante la realización de dicha prueba.

Identificador	
Fecha	
Responsable	
Descripción	
Resultado	
Descripción de los fallos	

Tabla 187. Plantilla de informe de pruebas

## 8 Construcción del sistema de Información

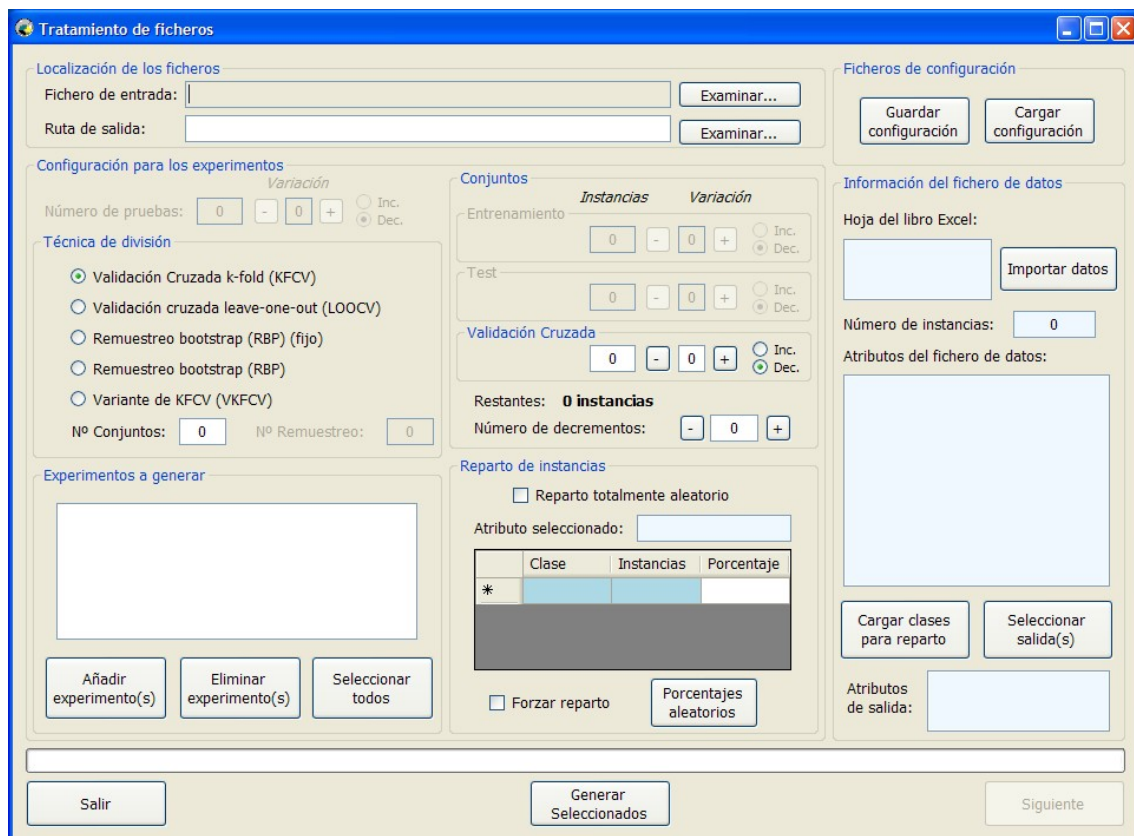
En este proceso se generará el código necesario para implementar el sistema además de la ejecución de pruebas.

### 8.1 Generación del código

Durante la fase de implementación se ha generado el código necesario para el funcionamiento del sistema desarrollado en el proyecto.

La aplicación desarrollada utiliza principalmente dos interfaces.

La **interfaz de tratamiento** permite configurar todas las opciones de los experimentos que se quieren preparar, así como importar los datos del fichero de entrada en formato de Microsoft Excel. A continuación se muestra la interfaz de tratamiento.



The interface is titled "Tratamiento de ficheros" and is divided into several sections:

- Localización de los ficheros:** Includes fields for "Fichero de entrada:" and "Ruta de salida:" with "Examinar..." buttons.
- Ficheros de configuración:** Includes "Guardar configuración" and "Cargar configuración" buttons.
- Configuración para los experimentos:**
  - Número de pruebas:** A numeric field with increment/decrement buttons and radio buttons for "Inc." and "Dec.".
  - Técnica de división:** Radio buttons for "Validación Cruzada k-fold (KFCV)", "Validación cruzada leave-one-out (LOOCV)", "Remuestreo bootstrap (RBP) (fijo)", "Remuestreo bootstrap (RBP)", and "Variante de KFCV (VKFCV)".
  - Nº Conjuntos:** 0 and **Nº Remuestreo:** 0.
- Conjuntos:**
  - Entrenamiento:** Numeric field with increment/decrement buttons and radio buttons for "Inc." and "Dec.".
  - Test:** Numeric field with increment/decrement buttons and radio buttons for "Inc." and "Dec.".
  - Validación Cruzada:** Numeric field with increment/decrement buttons and radio buttons for "Inc." and "Dec.".
  - Restantes:** 0 instancias.
  - Número de decrementos:** Numeric field with increment/decrement buttons.
- Reparto de instancias:**
  - ☐ Reparto totalmente aleatorio.
  - Atributo seleccionado:** Text field.
  - | Clase | Instancias | Porcentaje |
|-------|------------|------------|
| *     |            |            |
  - ☐ Forzar reparto.
  - Porcentajes aleatorios.
- Experimentos a generar:** A large empty box with buttons "Añadir experimento(s)", "Eliminar experimento(s)", and "Seleccionar todos".
- Información del fichero de datos:**
  - Hoja del libro Excel:** Text field with "Importar datos" button.
  - Número de instancias:** 0.
  - Atributos del fichero de datos:** Large empty box.
  - Buttons: "Cargar clases para reparto" and "Seleccionar salida(s)".
  - Atributos de salida:** Text field.

At the bottom, there are buttons for "Salir", "Generar Seleccionados", and "Siguiente".

Figura 29. Interfaz Tratamiento



La funcionalidad de esta interfaz consiste en acceder a un fichero de datos en formato Excel (ver Anexo B: Formato para ficheros de entrada), y preparar los experimentos en función de la configuración seleccionada por el usuario. La preparación de los experimentos se realiza generando los ficheros correspondientes de entrenamiento, validación y test en el formato adecuado para las redes de neuronas artificiales. Estos ficheros deben ser generados de acuerdo a las cantidades de instancias y atributos seleccionados por el usuario. Asimismo, el reparto de las instancias disponibles se realiza en función de la técnica de división elegida (ver 12.4.1. Técnica de división).

La interfaz de entrenamiento utiliza los ficheros para el entrenamiento, validación y producción de las redes neuronales artificiales. A continuación se muestra la interfaz de entrenamiento

Figura 30. Interfaz Entrenamiento



La funcionalidad de esta interfaz consiste en acceder cargar e identificar la tipología de una red neuronal y entrenar los experimentos seleccionados en las redes seleccionadas bajo un criterio de parada. Asimismo permite indicar si se desea que se generen los ficheros de evaluación y resumen del entrenamiento.

## **8.2 Ejecución de las pruebas unitarias**

La ejecución de las pruebas unitarias se realizará durante la generación las clases en el código.

## **8.3 Ejecución de las pruebas de Integración**

Con esta actividad se documentará la preparación, realización y evaluación de las diferentes pruebas definidas en el estudio del sistema de Información para verificar la correcta integración entre los componentes que se han codificado.

Se comprueba que todas las pruebas de integración definidas en el plan, son evaluadas con resultado satisfactorio.

## **8.4 Ejecución de las pruebas de sistema**

El objetivo de esta actividad es la ejecución de las diferentes pruebas que garantizan el correcto funcionamiento del sistema y prevén la aceptación por parte del cliente.

A continuación mostraremos los resultados de las pruebas realizadas descritas en el plan de pruebas en ambos dominios.

### **8.4.1 Dominio A**

A continuación mostramos las pruebas realizadas en el dominio de wine quality (calidad del vino).



En la siguiente figura podemos observar los resultados del fichero resumen resultante de la realización del plan de pruebas A anteriormente definido.

TIPOLOGÍA	PRUEBAS	TRAIN	TEST	ACIERTOS	FALLOS	TOTAL	%ACIERTOS	%FALLOS
Multilayer Perceptron (1_mlp.nsb)	200	1000	400	47223	32777	80000	59.029%	40.971%
Jordam-Elman (2_jordan.nsb)	200	1000	400	46094	33906	80000	57.618%	42.383%
Radial basis function (3_rbf.nsb)	200	1000	400	45846	34154	80000	57.308%	42.693%
Self-organizing feature map (4_map.nsb)	200	1000	400	46732	33268	80000	58.415%	41.585%
Recurrent Network (5_recurrent.nsb)	200	1000	400	46289	33711	80000	57.861%	42.139%
Support Vector Machine (6_vector.nsb)	200	1000	400	28544	51456	80000	35.680%	64.320%
Multilayer Perceptron (1_mlp.nsb)	150	1000	400	35417	24583	60000	59.028%	40.972%
Jordam-Elman (2_jordan.nsb)	150	1000	400	34634	25366	60000	57.723%	42.277%
Radial basis function (3_rbf.nsb)	150	1000	400	34495	25505	60000	57.492%	42.508%
Self-organizing feature map (4_map.nsb)	150	1000	400	35198	24802	60000	58.663%	41.337%
Recurrent Network (5_recurrent.nsb)	150	1000	400	34740	25260	60000	57.900%	42.100%
Support Vector Machine (6_vector.nsb)	150	1000	400	21453	38547	60000	35.755%	64.245%
Multilayer Perceptron (1_mlp.nsb)	100	1000	400	23472	16528	40000	58.680%	41.320%
Jordam-Elman (2_jordan.nsb)	100	1000	400	22933	17067	40000	57.333%	42.668%
Radial basis function (3_rbf.nsb)	100	1000	400	22862	17138	40000	57.155%	42.845%
Self-organizing feature map (4_map.nsb)	100	1000	400	23277	16723	40000	58.193%	41.808%
Recurrent Network (5_recurrent.nsb)	100	1000	400	23022	16978	40000	57.555%	42.445%
Support Vector Machine (6_vector.nsb)	100	1000	400	14386	25614	40000	35.965%	64.035%
Multilayer Perceptron (1_mlp.nsb)	50	1000	400	11701	8299	20000	58.505%	41.495%
Jordam-Elman (2_jordan.nsb)	50	1000	400	11552	8448	20000	57.760%	42.240%
Radial basis function (3_rbf.nsb)	50	1000	400	11498	8502	20000	57.490%	42.510%
Self-organizing feature map (4_map.nsb)	50	1000	400	11636	8364	20000	58.180%	41.820%
Recurrent Network (5_recurrent.nsb)	50	1000	400	11558	8442	20000	57.790%	42.210%
Support Vector Machine (6_vector.nsb)	50	1000	400	7109	12891	20000	35.545%	64.455%

Figura 31. Fichero resumen plan de pruebas A del dominio A

En la siguiente tabla se muestra la influencia que tiene el rango de valores usado en cada variable para el entrenamiento.

TIPOLOGÍA	PRUEBAS	TRAIN	TEST	fixed_acidity	volatile_acidity	citric_acid	residual_sugar	chlorides	free_sulfur_dioxide	total_sulfur_dioxide	density	pH	sulphates	alcohol
Multilayer Perceptron (1_mlp.nsb)	200	1000	400	0,846153846	0,352112676	0,9318182	0,398305085	0,587413	0,152380952	0,828125	0,6061	1	0,6125	0,8065
Jordam-Elman (2_jordan.nsb)	200	1000	400	0,879120879	0,718309859	0,9204545	0,745762712	0,51049	0,2	0,8359375	0,6818	1	0,99375	0,7742
Radial basis function (3_rbf.nsb)	200	1000	400	0,857142857	0,690140845	0,7840909	0,5	0,545455	0,333333333	0,7578125	0,4773	1	0,7125	0,8172
Self-organizing feature map (4_map.nsb)	200	1000	400	0,791208791	0,718309859	0,9318182	0,288135593	0,594406	0,523809524	0,734375	0,5303	1	0,7625	0,5484
Recurrent Network (5_recurrent.nsb)	200	1000	400	0,818681319	0,732394366	0,8522727	0,805084746	0,51049	0,142857143	0,7421875	0,6364	1	0,875	0,8387
Support Vector Machine (6_vector.nsb)	200	1000	400	0,813186813	0,830985915	0,8522727	0,389830508	0,944056	0,219047619	0,796875	0,1742	0	0,675	0,7312
Multilayer Perceptron (1_mlp.nsb)	150	1000	400	0,913354839	0,294117647	1	0,103448276	0,59434	0,115789474	0,818181818	0,4691	1	0,542553	0,7808
Jordam-Elman (2_jordan.nsb)	150	1000	400	0,935483871	0,623529412	1	0,632183908	0,424528	0,105263158	0,893939394	0,5802	1	1	0,7808
Radial basis function (3_rbf.nsb)	150	1000	400	0,89516129	0,494117647	0,9333333	0,448275862	0,45283	0,210526316	0,909090909	0,3704	1	0,744681	0,7671
Self-organizing feature map (4_map.nsb)	150	1000	400	0,806451613	0,682352941	1	0,103448276	0,424528	0,621052632	0,848484848	0,4691	1	0,702128	0,589
Recurrent Network (5_recurrent.nsb)	150	1000	400	0,85483871	0,658823529	0,9333333	0,873663218	0,424528	0,073684211	0,893939394	0,6914	1	0,851064	0,8219
Support Vector Machine (6_vector.nsb)	150	1000	400	0,887096774	0,811764706	0,9333333	0,402298851	0,424528	0,231578947	0,954545455	0,0617	1	0,62766	0,7808
Multilayer Perceptron (1_mlp.nsb)	100	1000	400	0,907692308	0,493506494	0,9166667	0,49382716	0,511905	0,12	0,885714286	0,6111	1	0,507246	0,7143
Jordam-Elman (2_jordan.nsb)	100	1000	400	0,907692308	0,753246753	0,9722222	0,716049383	0,428571	0,06	0,814285714	0,6296	1	1	0,75
Radial basis function (3_rbf.nsb)	100	1000	400	0,907692308	0,649350649	0,9166667	0,641975309	0,452381	0,16	0,8	0,4074	0	0,695652	0,7321
Self-organizing feature map (4_map.nsb)	100	1000	400	0,738461538	0,818181818	1	0,209876543	0,428571	0,72	0,814285714	0,5185	1	0,695652	0,5179
Recurrent Network (5_recurrent.nsb)	100	1000	400	0,876923077	0,766233766	0,9444444	0,839506173	0,428571	0,06	0,8	0,7407	1	0,797101	0,8036
Support Vector Machine (6_vector.nsb)	100	1000	400	0,907692308	0,727272727	0,9166667	0,469135802	0,452381	0,12	0,828571429	0,037	0	0,623188	0,7321
Multilayer Perceptron (1_mlp.nsb)	50	1000	400	0,794871795	0,357142857	0,9230769	0,875	0,525	0	0,894736842	0,8148	1	0,361111	0,9048
Jordam-Elman (2_jordan.nsb)	50	1000	400	0,871794872	0,714285714	1	0,75	0,525	0,058823529	0,842105263	0,7407	1	1	0,9048
Radial basis function (3_rbf.nsb)	50	1000	400	0,794871795	0,392857143	0,9230769	0,708333333	0,55	0,294117647	0,842105263	0,6296	1	0,722222	0,9048
Self-organizing feature map (4_map.nsb)	50	1000	400	0,641025641	0,642857143	1	0,125	0,525	0,470588235	0,894736842	0,7037	1	0,638889	0,5238
Recurrent Network (5_recurrent.nsb)	50	1000	400	0,743589744	0,642857143	0,9230769	0,958333333	0,525	0,058823529	0,842105263	0,8889	1	0,805556	0,9524
Support Vector Machine (6_vector.nsb)	50	1000	400	0,794871795	0,678571429	0,9230769	0,583333333	0,55	0,411764706	0,842105263	0,1852	1	0,527778	0,9048

Figura 32. Influencia del fichero resumen plan de pruebas B del dominio A





En la siguiente figura podemos observar los resultados del fichero resumen resultante de la realización del plan de pruebas B anteriormente definido.

	A	B	C	D	E	F	G	H	I
1	TIPOLOGÍA	PRUEBAS	TRAIN	TEST	ACIERTOS	FALLOS	TOTAL	%ACIERTOS	%FALLOS
2	Multilayer Perceptron (1_mlp.nsb)	200	800	600	70907	49093	120000	59.089%	40.911%
3	Jordam-Elman (2_jordan.nsb)	200	800	600	69146	50854	120000	57.622%	42.378%
4	Radial basis function (3_rbf.nsb)	200	800	600	68161	51839	120000	56.801%	43.199%
5	Self-organizing feature map (4_map.nsb)	200	800	600	68872	51128	120000	57.393%	42.607%
6	Recurrent Network (5_recurrent.nsb)	200	800	600	69644	50356	120000	58.037%	41.963%
7	Support Vector Machine (6_vector.nsb)	200	800	600	44566	75434	120000	37.138%	62.862%
8	Multilayer Perceptron (1_mlp.nsb)	150	800	600	53270	36730	90000	59.189%	40.811%
9	Jordam-Elman (2_jordan.nsb)	150	800	600	50363	39637	90000	55.959%	44.041%
10	Radial basis function (3_rbf.nsb)	150	800	600	50362	39638	90000	55.958%	44.042%
11	Self-organizing feature map (4_map.nsb)	150	800	600	51975	38025	90000	57.750%	42.250%
12	Recurrent Network (5_recurrent.nsb)	150	800	600	52136	37864	90000	57.929%	42.071%
13	Support Vector Machine (6_vector.nsb)	150	800	600	35407	54593	90000	39.341%	60.659%
14	Multilayer Perceptron (1_mlp.nsb)	100	800	600	34973	25027	60000	58.288%	41.712%
15	Jordam-Elman (2_jordan.nsb)	100	800	600	34246	25754	60000	57.077%	42.923%
16	Radial basis function (3_rbf.nsb)	100	800	600	33691	26309	60000	56.152%	43.848%
17	Self-organizing feature map (4_map.nsb)	100	800	600	34909	25091	60000	58.182%	41.818%
18	Recurrent Network (5_recurrent.nsb)	100	800	600	34717	25283	60000	57.862%	42.138%
19	Support Vector Machine (6_vector.nsb)	100	800	600	25369	34631	60000	42.282%	57.718%
20	Multilayer Perceptron (1_mlp.nsb)	50	800	600	17508	12492	30000	58.360%	41.640%
21	Jordam-Elman (2_jordan.nsb)	50	800	600	17143	12857	30000	57.143%	42.857%
22	Radial basis function (3_rbf.nsb)	50	800	600	16748	13252	30000	55.827%	44.173%
23	Self-organizing feature map (4_map.nsb)	50	800	600	17480	12520	30000	58.267%	41.733%
24	Recurrent Network (5_recurrent.nsb)	50	800	600	17338	12662	30000	57.793%	42.207%
25	Support Vector Machine (6_vector.nsb)	50	800	600	12925	17075	30000	43.083%	56.917%

Figura 33. Fichero resumen plan de pruebas B del dominio A

En la siguiente tabla se muestra la influencia que tiene el rango de valores usado en cada variable para el entrenamiento.

28	TIPOLOGIA	PRUEBAS	TRAIN	TEST	fixed_acidity	volatile_acidity	citric_acid	residual_sugar	chlorides	free_sulfur_dioxide	total_sulfur_dioxide	density	pH	sulphates	alcohol
29	Multilayer Perceptron (1_mlp.nsb)	200	800	600	0.781481481	0.679487179	0.819549	0.59375	0.501887	0.374358974	0.792270531	0.7418	1	0.691756	0.6962
30	Jordam-Elman (2_jordan.nsb)	200	800	600	0.785185185	0.427350427	0.691729	0.6875	0.569811	0.246153846	0.830917874	0.5984	0	0.831541	0.6646
31	Radial basis function (3_rbf.nsb)	200	800	600	0.785185185	0.435897436	0.917293	0.321428571	0.501887	0.302564103	0.888888889	0.3197	1	0.799283	0.7532
32	Self-organizing feature map (4_map.nsb)	200	800	600	0.837037037	0.427350427	0.827068	0.348214286	0.679245	0.276923077	0.758454106	0.1803	0	0.770609	0.6835
33	Recurrent Network (5_recurrent.nsb)	200	800	600	0.766666667	0.521367521	0.849624	0.5625	0.490566	0.158974359	0.787439614	0.5082	0	0.878136	0.7278
34	Support Vector Machine (6_vector.nsb)	200	800	600	0.751851852	0.709401709	0.819549	0.3125	0.822642	0.435897436	0.855072464	0.2254	0	0.756272	0.6392
35	Multilayer Perceptron (1_mlp.nsb)	150	800	600	0.857777778	0.921212121	0.953271	0.392265193	0.509901	0.517006803	0.851190476	0.4058	1	0.781659	0.8714
36	Jordam-Elman (2_jordan.nsb)	150	800	600	0.804444444	0.660606061	0.841121	0.480662983	0.569307	0.25170068	0.797619048	0.4928	0	0.868996	0.6643
37	Radial basis function (3_rbf.nsb)	150	800	600	0.782222222	0.690909091	0.934579	0.209944751	0.49505	0.585034014	0.839285714	0.4106	1	0.812227	0.8357
38	Self-organizing feature map (4_map.nsb)	150	800	600	0.791111111	0.539393939	0.850467	0.320441989	0.79703	0.31292517	0.726190476	0.2077	0	0.829694	0.6643
39	Recurrent Network (5_recurrent.nsb)	150	800	600	0.768888889	0.539393939	0.859813	0.674033149	0.554455	0.054421769	0.744047619	0.6425	0	0.930131	0.6571
40	Support Vector Machine (6_vector.nsb)	150	800	600	0.453333333	0.660606061	0.719626	0.303867403	0.519802	0.448979592	0.93452381	0.4976	1	0.615721	0.4786
41	Multilayer Perceptron (1_mlp.nsb)	100	800	600	0.920353982	0.62295082	0.952381	0.555555556	0	0.210526316	0.918367347	0.418	1	0.909091	0.7653
42	Jordam-Elman (2_jordan.nsb)	100	800	600	0.867256637	0.459016393	0.936508	0.623931624	0.608	0.210526316	0.836734694	0.5328	1	0.975207	0.6122
43	Radial basis function (3_rbf.nsb)	100	800	600	0.769911504	0.909836066	0.84127	0.188034188	0.488	0.50877193	0.846938776	0.5082	1	0.793388	0.8367
44	Self-organizing feature map (4_map.nsb)	100	800	600	0.734513274	0.614754098	0.857143	0.367521368	0.896	0.350877193	0.806122449	0.0984	0	0.892562	0.6837
45	Recurrent Network (5_recurrent.nsb)	100	800	600	0.787610619	0.56557377	0.920635	0.692307692	0.656	0.043859649	0.806122449	0.5656	0	0.966942	0.602
46	Support Vector Machine (6_vector.nsb)	100	800	600	0.743362832	0.680327869	0.857143	0.076923077	0.808	0.078947368	0.87755102	0.3525	0	0.809917	0.5714
47	Multilayer Perceptron (1_mlp.nsb)	50	800	600	0.865671642	0.603448276	1	0.553846154	0.820896	0	0.680555556	0.3974	1	0.934426	0.9688
48	Jordam-Elman (2_jordan.nsb)	50	800	600	0.835820896	0.413793103	0.96875	0.676923077	0.656716	0.134328358	0.75	0.5385	1	0.983607	0.75
49	Radial basis function (3_rbf.nsb)	50	800	600	0.776119403	0.948275862	0.78125	0.215384615	0.477612	0.611940299	0.833333333	0.5513	1	0.770492	0.9688
50	Self-organizing feature map (4_map.nsb)	50	800	600	0.76119403	0.482758621	0.8125	0.353846154	0.880597	0.358208955	0.708333333	0.1923	0	0.885246	0.7188
51	Recurrent Network (5_recurrent.nsb)	50	800	600	0.820895522	0.431034483	0.96875	0.692307692	0.716418	0.014925373	0.694444444	0.6154	0	0.918033	0.7188
52	Support Vector Machine (6_vector.nsb)	50	800	600	0.76119403	0.655172414	0.8125	0.123076923	0.791045	0.029850746	0.791666667	0.4744	0	0.819672	0.625

Figura 34. Influencia del fichero resumen plan de pruebas B del dominio A



## 8.4.2 Dominio B

A continuación mostramos las pruebas realizadas en el dominio del impacto balístico:

En este caso, al ser experimentos de regresión, se calculará en los ficheros de evaluación y resumen el factor de correlación para las variables de salida.

En la siguiente figura podemos observar los resultados del fichero resumen resultante de la realización del plan de pruebas A anteriormente definido.

	A	B	C	D	E	F	G
1	TIPOLOGÍA RED	PRUEBAS	TRAIN	TEST	TOTAL	Media (Mi-Mf)/Mi (%)	Media Vres (m/s)
2	Multilayer Perceptron (1_mlp.nsb)	300	110	35	10500	0,750032857	0,928691813
3	Jordam-Elman (2_jordan.nsb)	300	110	35	10500	0,690052064	0,903493757
4	Radial basis function (3_rbf.nsb)	300	110	35	10500	0,691601133	0,869182575
5	Self-organizing feature map (4_map.nsb)	300	110	35	10500	0,725042744	0,906977811
6	Recurrent Network (5_recurrent.nsb)	300	110	35	10500	0,582195481	0,781259989
7	Multilayer Perceptron (1_mlp.nsb)	250	110	35	8750	0,726604754	0,922616897
8	Jordam-Elman (2_jordan.nsb)	250	110	35	8750	0,671654964	0,916099298
9	Radial basis function (3_rbf.nsb)	250	110	35	8750	0,656192178	0,876438302
10	Self-organizing feature map (4_map.nsb)	250	110	35	8750	0,708016302	0,911278234
11	Recurrent Network (5_recurrent.nsb)	250	110	35	8750	0,606234983	0,822303791
12	Multilayer Perceptron (1_mlp.nsb)	200	110	35	7000	0,75769359	0,919220515
13	Jordam-Elman (2_jordan.nsb)	200	110	35	7000	0,709141373	0,920388811
14	Radial basis function (3_rbf.nsb)	200	110	35	7000	0,659892891	0,871566755
15	Self-organizing feature map (4_map.nsb)	200	110	35	7000	0,715534557	0,90992924
16	Recurrent Network (5_recurrent.nsb)	200	110	35	7000	0,628547974	0,834481678
17	Multilayer Perceptron (1_mlp.nsb)	150	110	35	5250	0,789434544	0,928402971
18	Jordam-Elman (2_jordan.nsb)	150	110	35	5250	0,709852938	0,921064093
19	Radial basis function (3_rbf.nsb)	150	110	35	5250	0,660512554	0,869085229
20	Self-organizing feature map (4_map.nsb)	150	110	35	5250	0,718211899	0,914142525
21	Recurrent Network (5_recurrent.nsb)	150	110	35	5250	0,63020488	0,850883602
22	Multilayer Perceptron (1_mlp.nsb)	100	110	35	3500	0,795644847	0,926434225
23	Jordam-Elman (2_jordan.nsb)	100	110	35	3500	0,719584533	0,920887193
24	Radial basis function (3_rbf.nsb)	100	110	35	3500	0,682424579	0,877253006
25	Self-organizing feature map (4_map.nsb)	100	110	35	3500	0,762629564	0,928108028
26	Recurrent Network (5_recurrent.nsb)	100	110	35	3500	0,653647328	0,855278501
27	Multilayer Perceptron (1_mlp.nsb)	50	110	35	1750	0,8183036	0,926483973
28	Jordam-Elman (2_jordan.nsb)	50	110	35	1750	0,724116211	0,920829108
29	Radial basis function (3_rbf.nsb)	50	110	35	1750	0,702489659	0,871741461
30	Self-organizing feature map (4_map.nsb)	50	110	35	1750	0,777611347	0,92214582
31	Recurrent Network (5_recurrent.nsb)	50	110	35	1750	0,652347366	0,852049051

Figura 35. Fichero resumen plan de pruebas A del dominio B

En la siguiente figura podemos observar los resultados del fichero resumen resultante de la realización del plan de pruebas B anteriormente definido.

	A	B	C	D	E	F	G
1	TIPOLOGÍA RED	PRUEBAS	TRAIN	TEST	TOTAL	Media (Mi-Mf)/Mi (%)	Media Vres (m/s)
2	Multilayer Perceptron (1_mlp.nsb)	300	80	50	15000	0,732624165	0,921314487
3	Jordam-Elman (2_jordan.nsb)	300	80	50	15000	0,75400465	0,868849991
4	Radial basis function (3_rbf.nsb)	300	80	50	15000	0,650970017	0,877881949
5	Self-organizing feature map (4_map.nsb)	300	80	50	15000	0,694450487	0,858423203
6	Recurrent Network (5_recurrent.nsb)	300	80	50	15000	0,631414701	0,806605443
7	Multilayer Perceptron (1_mlp.nsb)	250	80	50	12500	0,746346528	0,919388255
8	Jordam-Elman (2_jordan.nsb)	250	80	50	12500	0,734093711	0,876356137
9	Radial basis function (3_rbf.nsb)	250	80	50	12500	0,650801257	0,879664072
10	Self-organizing feature map (4_map.nsb)	250	80	50	12500	0,675106589	0,867881238
11	Recurrent Network (5_recurrent.nsb)	250	80	50	12500	0,632669452	0,811226657
12	Multilayer Perceptron (1_mlp.nsb)	200	80	50	10000	0,729490231	0,909815762
13	Jordam-Elman (2_jordan.nsb)	200	80	50	10000	0,735054957	0,876395237
14	Radial basis function (3_rbf.nsb)	200	80	50	10000	0,662915058	0,854751424
15	Self-organizing feature map (4_map.nsb)	200	80	50	10000	0,668036214	0,869748455
16	Recurrent Network (5_recurrent.nsb)	200	80	50	10000	0,632924084	0,80439992
17	Multilayer Perceptron (1_mlp.nsb)	150	80	50	7500	0,738816058	0,907210285
18	Jordam-Elman (2_jordan.nsb)	150	80	50	7500	0,74237458	0,874346943
19	Radial basis function (3_rbf.nsb)	150	80	50	7500	0,63723191	0,850223429
20	Self-organizing feature map (4_map.nsb)	150	80	50	7500	0,664395182	0,867367379
21	Recurrent Network (5_recurrent.nsb)	150	80	50	7500	0,62278416	0,820919154
22	Multilayer Perceptron (1_mlp.nsb)	100	80	50	5000	0,717472735	0,915884773
23	Jordam-Elman (2_jordan.nsb)	100	80	50	5000	0,711510825	0,875186603
24	Radial basis function (3_rbf.nsb)	100	80	50	5000	0,62969421	0,85061917
25	Self-organizing feature map (4_map.nsb)	100	80	50	5000	0,650310415	0,860588929
26	Recurrent Network (5_recurrent.nsb)	100	80	50	5000	0,596701691	0,827266477
27	Multilayer Perceptron (1_mlp.nsb)	50	80	50	2500	0,741977098	0,928344607
28	Jordam-Elman (2_jordan.nsb)	50	80	50	2500	0,739849895	0,880232419
29	Radial basis function (3_rbf.nsb)	50	80	50	2500	0,649420224	0,864265394
30	Self-organizing feature map (4_map.nsb)	50	80	50	2500	0,666092839	0,873993998
31	Recurrent Network (5_recurrent.nsb)	50	80	50	2500	0,621893222	0,833418489

Figura 36. Fichero resumen plan de pruebas A del dominio B



## **9 Conclusiones**

La realización de este proyecto ha conseguido los objetivos marcados al inicio del mismo.

En primer lugar, se ha creado una aplicación que desarrolla la funcionalidad deseada. Permite realizar informes sobre las simulaciones realizadas en redes neuronales a partir de un grupo de datos. Además se ha conseguido que pueda utilizarse en el ámbito de cualquier dominio.

Para la realización del proyecto se ha utilizado METRICA V3, una metodología que ofrece una "infraestructura" básica con la que construir software desde la planificación para el desarrollo hasta el mantenimiento del sistema de información.

En cuanto a la planificación del proyecto, se han cumplido los plazos de entrega previstos.

Durante la ejecución de este proyecto se han aplicado los conocimientos adquiridos durante la carrera, además de adquirir nuevos conocimientos. Como el aprendizaje del lenguaje de programación.NET utilizado en este proyecto.

Finalmente, se ha comprobado la dificultad en cuanto a disciplina, esfuerzo y coste de realizar un proyecto de forma completa.



## 10 Bibliografía

[1] Hopfield, J.L.. *Neural networks and physical systems with emergent collective computational abilities*. The National Academy of Science USA, 79, 2554-2558, 1982.

[2] Ceballos Sierra, Francisco Javier. *Microsoft Visual Basic .Net : lenguaje y aplicaciones*. Madrid : Ra-Ma , 2006.

[3] González Carrasco, Israel. *Análisis, optimización y evaluación de modelos de redes de neuronas artificiales para la clasificación y predicción de impactos de alta velocidad sobre distintos materiales*. Leganés: Universidad Carlos III de Madrid, Departamento de Informática, 2010.

[4] Moreno Torres, Alejandro. *Aplicación de un perceptrón multicapa para resolución de problemas de impacto balístico*. Leganés: A. Moreno, 2008.

[5] Martín Muñoz, David. *Estudio de la viabilidad de aplicación de redes de neuronas artificiales para la resolución del problema del impacto sobre materiales*. Leganés: D. Martín, 2007.

[6] Díez Morcillo, Elisa. *Herramienta de gestión de conocimiento para un sistema experto basado en web*. Leganés: E. Díez, 2006.

[7] Fernández, D., Zaera, R., García-Crespo, A., Ruiz-Mezcua, B.. *Predicción mediante redes neuronales de la respuesta frente a impacto de alta velocidad de protecciones de acero*. Congreso de la Estructura de Acero 2004, La Coruña, Septiembre 2004.

[8] Ministerio de Administraciones Públicas, *"Métrica Versión 3"* [en línea]. Disponible en: <http://www.csi.map.es/csi/metrica3/>.

[9] NeuroDimension, Inc.. *NeuroSolutionsHelp* [en línea]. Disponible en: <http://www.neurosolutions.com/downloads/documentation.html>

[10] P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. *Modeling wine preferences by data mining from physicochemical properties* [en línea]. In Decision Support Systems, Elsevier, 47(4):547-553. ISSN: 0167-9236. Disponible en: <http://dx.doi.org/10.1016/j.dss.2009.05.016>



[11] Martín Muñoz, David. *Framework para la conexión de sistemas híbridos basados en redes de neuronas y sistemas expertos*. Leganés: D. Martín, 2009



## 11 Anexo A: Manual de usuario Aplicación Tratamiento y Entrenamiento en Redes Neuronales

A continuación se proporciona una manual para la utilización de las distintas funcionalidades disponibles en la aplicación. Esta manual describe al usuario el funcionamiento básico de la aplicación

### 11.1 Arranque de la aplicación

Para iniciar la aplicación sólo hay que hacer doble clic sobre el icono de la aplicación.

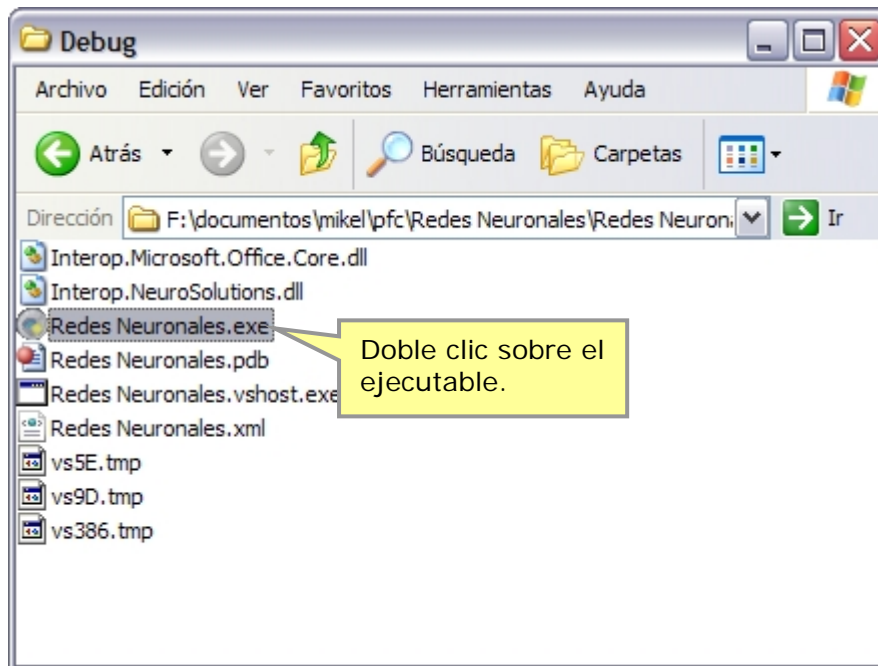


Figura 37. Arranque aplicación



Una vez iniciada la aplicación el usuario podrá visualizar el interfaz que permite el tratamiento de los datos.

Figura 38. Interfaz Tratamiento

## 11.2 Localización de ficheros

Una vez arrancada la aplicación, el primer paso es especificar la localización de los ficheros de datos. Esta operación debe incluir tanto el fichero de entrada original, como la ruta de salida para los ficheros generados por la aplicación.

### 11.2.1 Fichero de entrada

Para especificar el fichero de entrada, se debe seleccionar el fichero mediante el ratón utilizando el botón “Examinar...” situado junto al cuadro de texto “Fichero de entrada”, a través del cuadro de diálogo que mostrará la aplicación. Se debe tener en cuenta que el fichero seleccionado debe cumplir con los requisitos de formato establecidos por la aplicación (véase Anexo B: Formato para ficheros de entrada). La siguiente ilustración muestra el proceso:

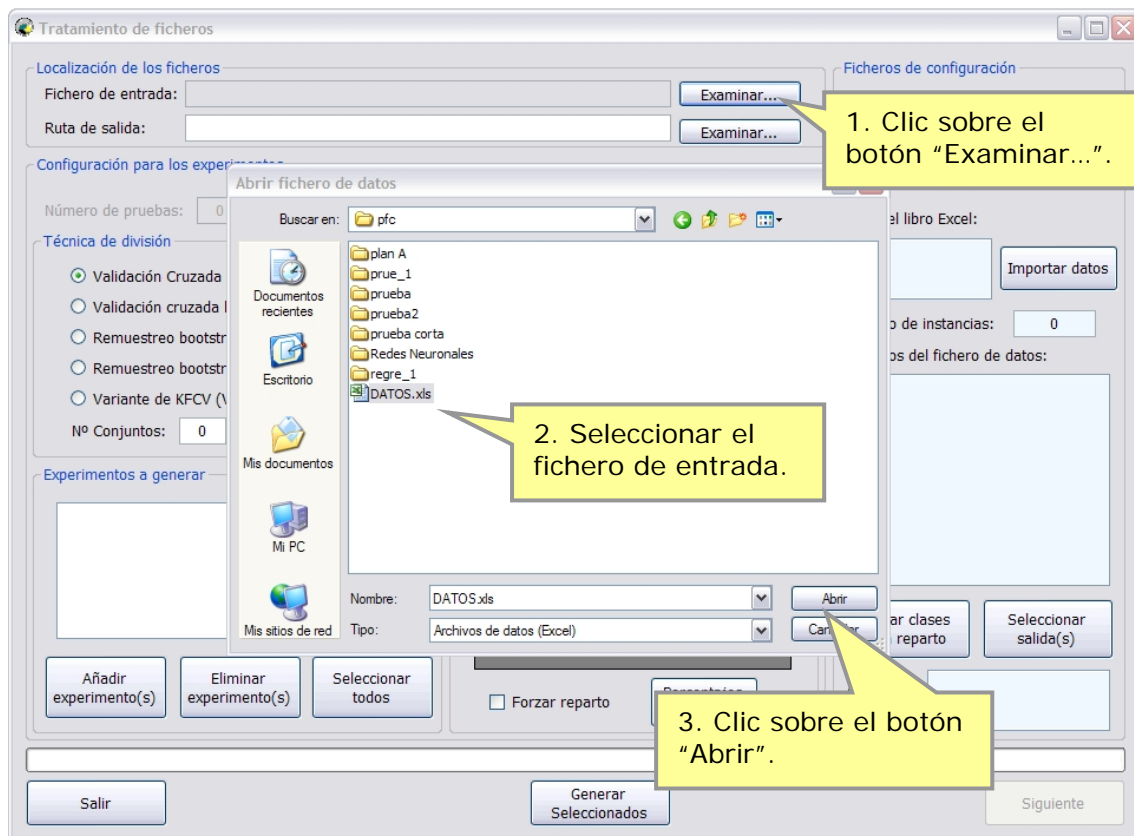


Figura 39. Seleccionar un fichero de entrada



## 11.2.2 Especificar una ruta de salida

Es necesario especificar una ruta de salida donde la aplicación almacenará los ficheros de entrenamiento, test y validación cruzada generados durante el proceso. Por un lado, se puede indicar la ruta en el cuadro de texto “Ruta de salida”, tal y como muestra la siguiente ilustración:

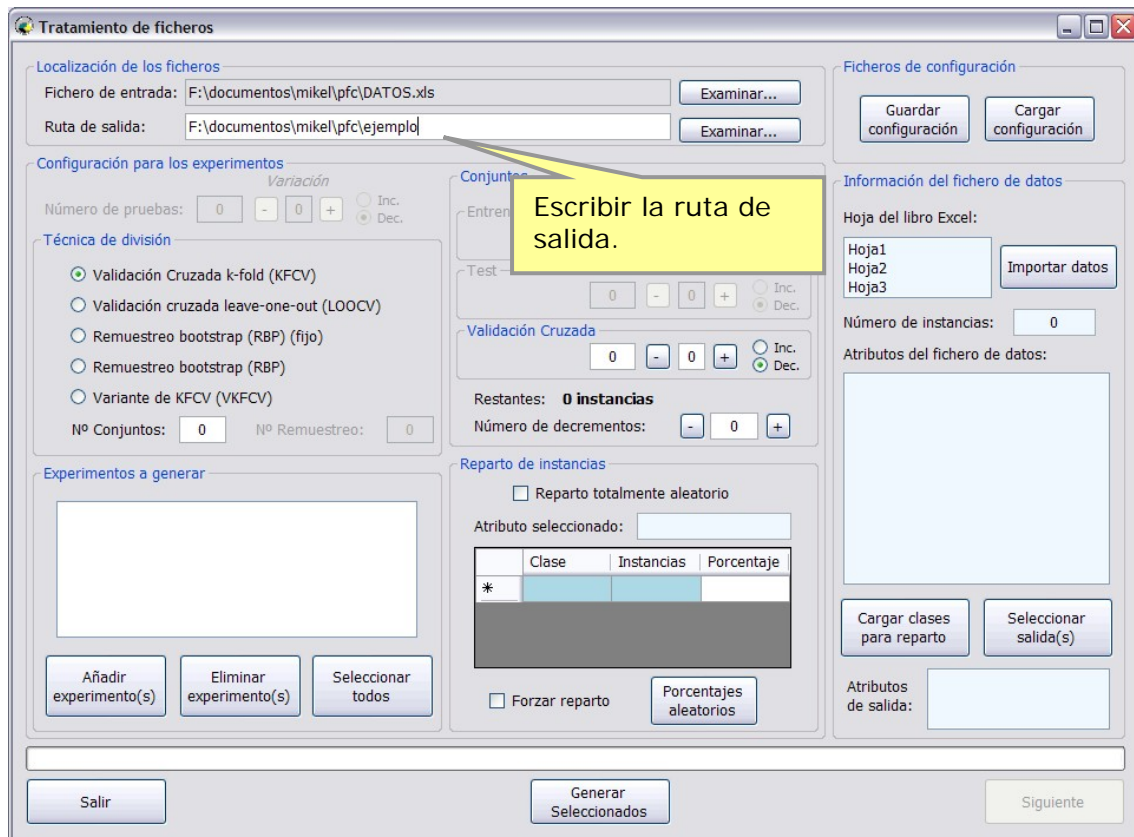


Figura 40. Selección ruta de salida manualmente

Otra opción es seleccionar la ruta de salida a través del cuadro de diálogo que mostrará la aplicación al hacer clic sobre el botón “Examinar...” situado junto al cuadro de texto “Ruta de salida”. La siguiente figura muestra el cuadro diálogo para seleccionar la ruta de salida:

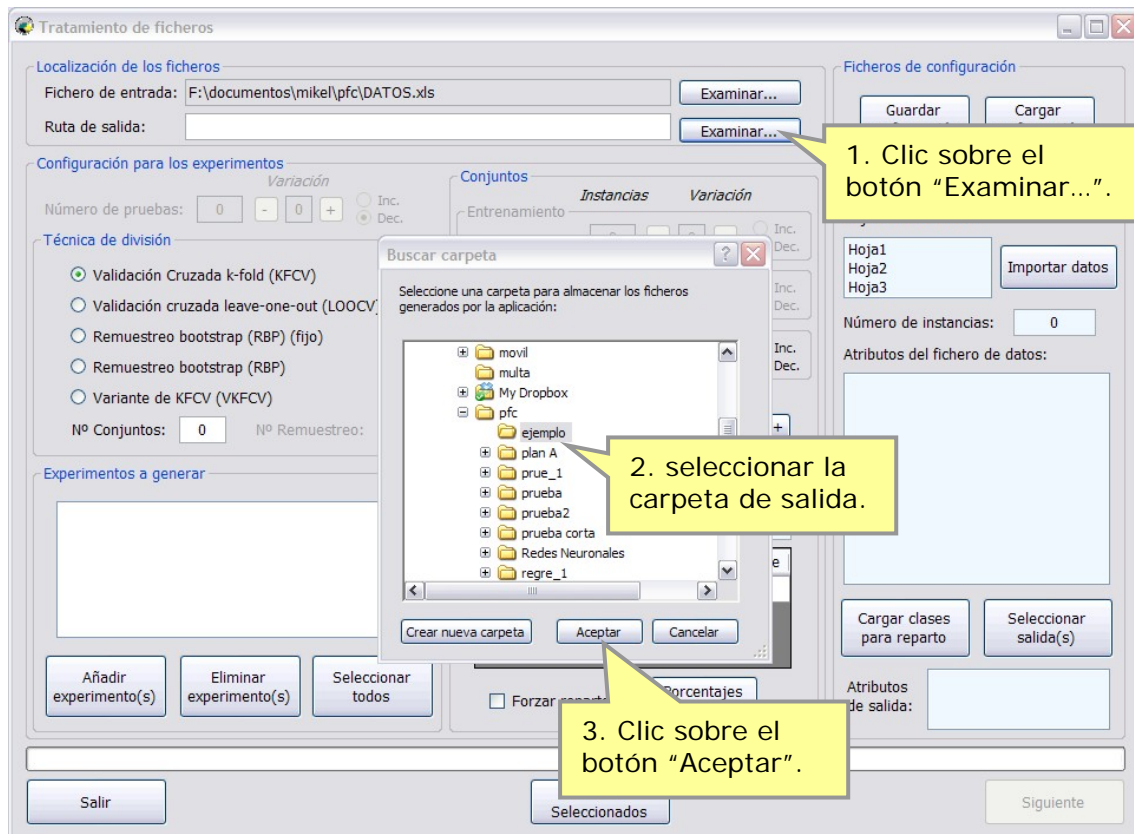


Figura 41. Selección ruta de salida

### 11.3 Importar datos

Una vez seleccionado un fichero de entrada con el formato adecuado, la aplicación extraerá la información necesaria. Para ello se debe especificar la hoja del libro Microsoft Excel donde se encuentran los datos y a continuación hacer clic sobre el botón “Importar datos”, tal y como se muestra a continuación:

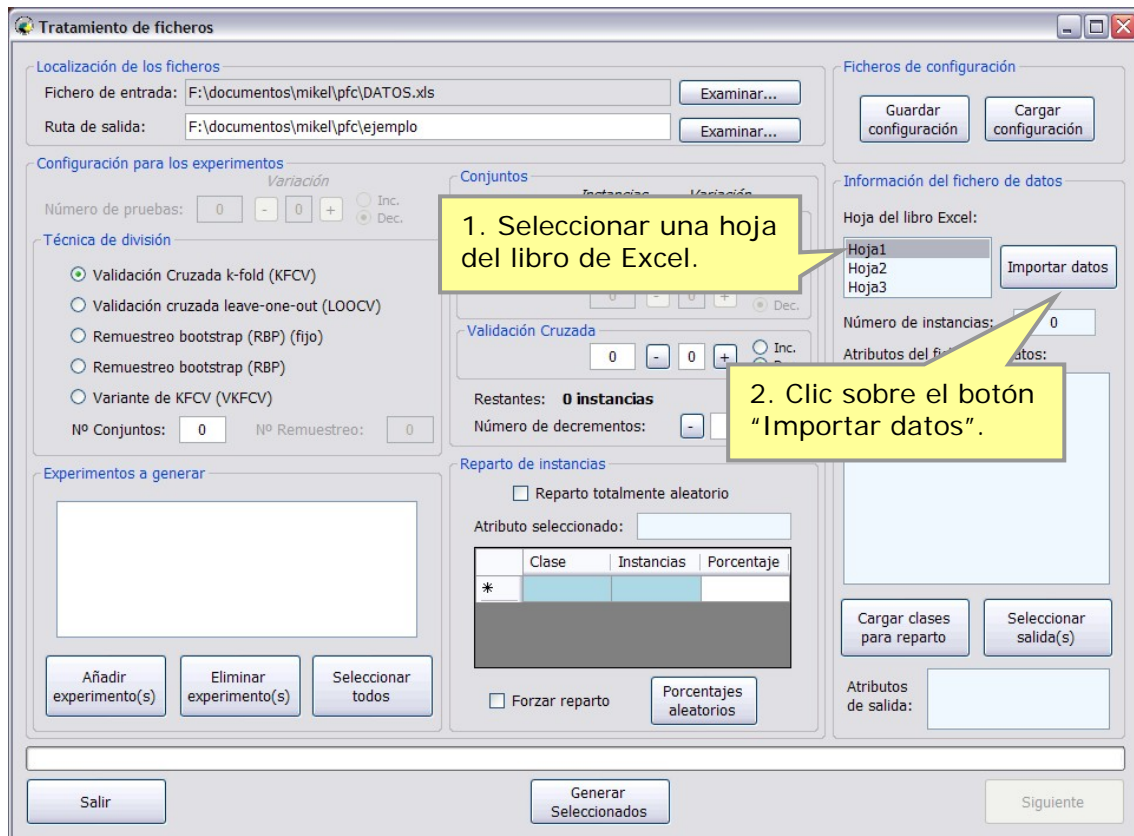


Figura 42. Importar datos

Al realizar esta operación se cargará el número de instancias encontradas en el cuadro de texto “Número de instancias” y la aplicación mostrará un cuadro de diálogo para la selección de atributos:

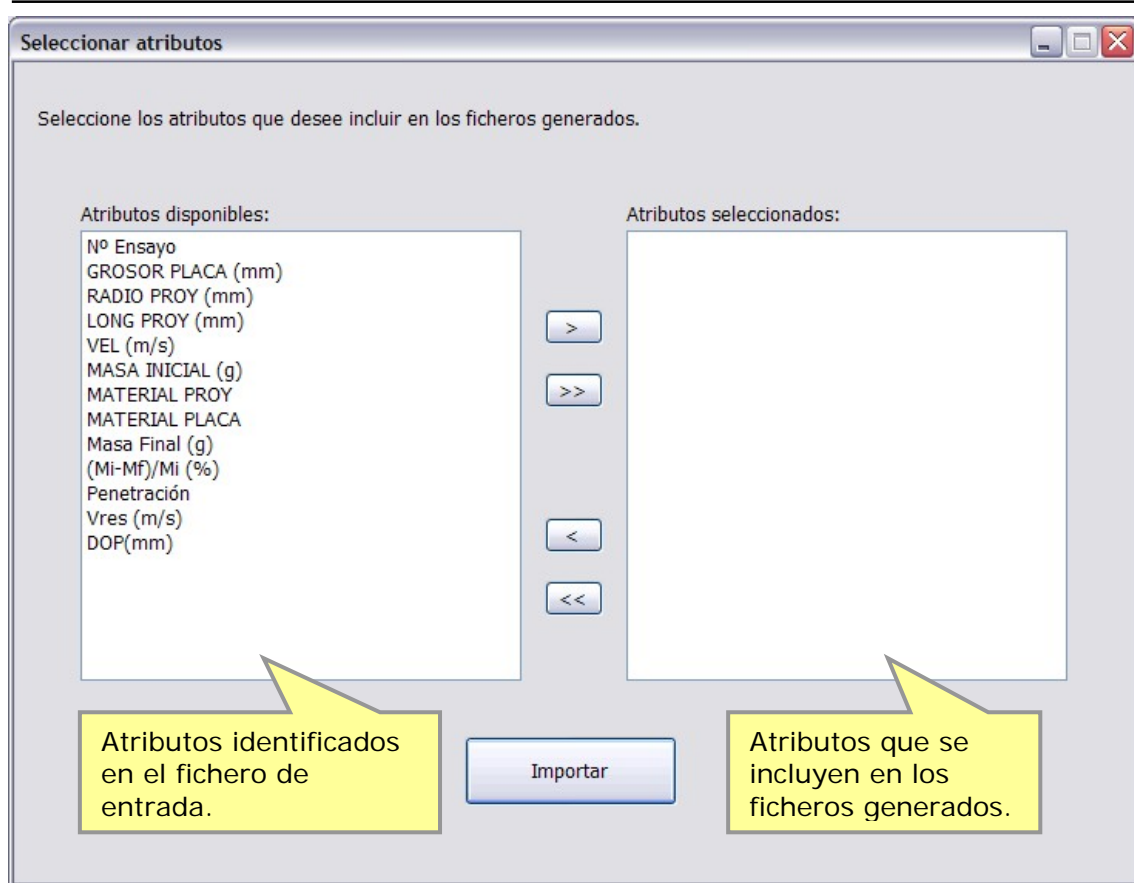


Figura 43. Seleccionar Atributos

Este cuadro de diálogo muestra en dos listas los atributos disponibles en el fichero de entrada, y los atributos seleccionados por el usuario para sus ficheros (inicialmente vacía). Para realizar la selección de atributos deseada se encuentran disponibles las siguientes operaciones:

- **Añadir atributo(s):** Mediante el botón ">" se añaden los atributos seleccionados en la lista de "Atributos disponibles" a la lista "Atributos seleccionados".
- **Eliminar atributo(s):** El botón "<" quita de la selección del usuario los atributos seleccionados previamente en la lista "Atributos seleccionados", volviendo a aparecer estos en la lista "Atributos disponibles".
- **Añadir todos:** Al hacer clic sobre el botón ">>" se añadirán todos los atributos que se encuentren en la lista "Atributos disponibles" a la lista "Atributos seleccionados".

- **Eliminar todos:** Si el usuario quiere eliminar todos los atributos de la lista "Atributos seleccionados" podrá hacerlo mediante un clic sobre el botón "<<".

Una vez terminada la selección de atributos deseados para los ficheros a generar se hace clic sobre el botón "Importar", y dichos atributos aparecerán en la lista de "Atributos del fichero de datos" de la interfaz de tratamiento de la aplicación.

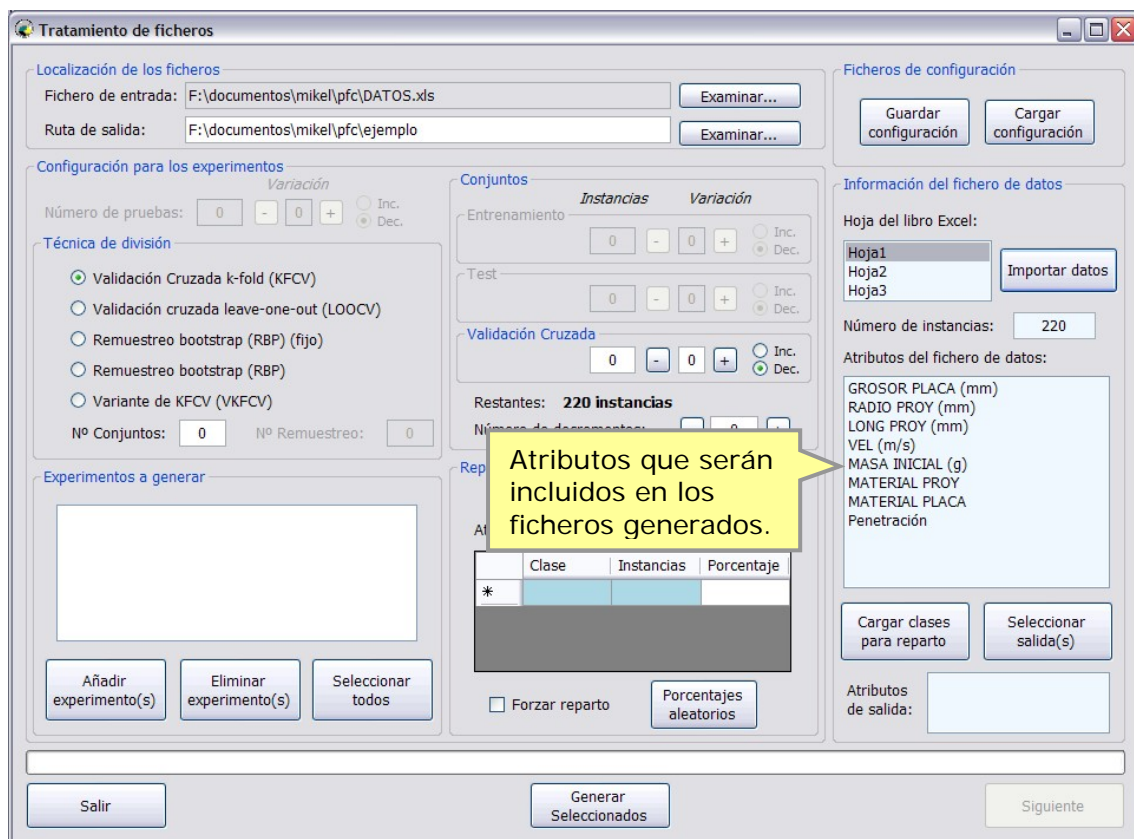


Figura 44. Atributos seleccionados

Tanto para clasificación como para regresión es necesario especificar que atributo o atributos serán los que la red de neuronas deberá predecir. Para ello se debe seleccionar uno o varios atributos cualesquiera de la lista de atributos seleccionados previamente y a continuación hacer clic sobre el botón “Seleccionar salida(s)”. Una vez seleccionados, los atributos aparecerán en el cuadro de texto “Atributos de salida”, tal y como se puede observar en la siguiente ilustración:

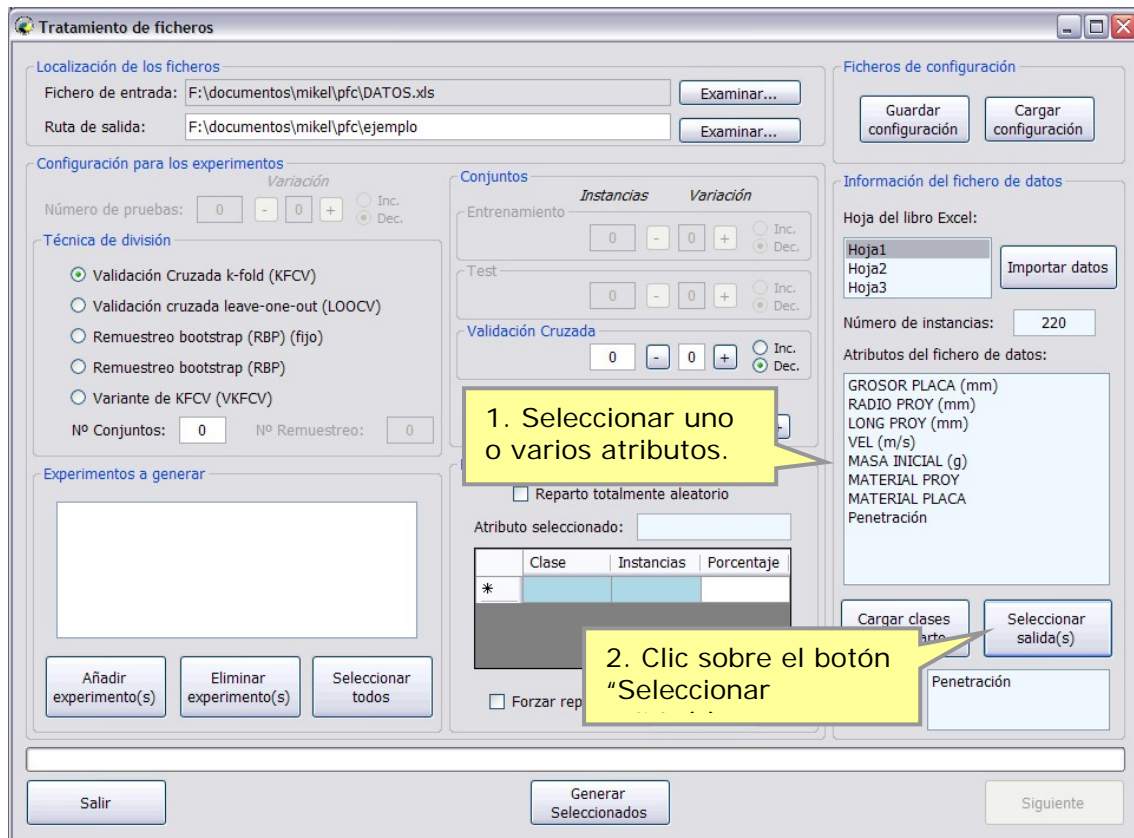


Figura 45. Selección atributos de salida



## 11.4 Configuración de experimentos

Una vez importados los datos del fichero de entrada y se han seleccionado los atributos del mismo, el usuario debe configurar cada uno de los experimentos que desee preparar. A continuación se detalla el proceso de configuración para un experimento nuevo.

### 11.4.1 Técnica de división

El primer parámetro de configuración que permite la aplicación es la técnica a emplear para la división del conjunto de instancias de partida para conformar los conjuntos de entrenamiento, test y validación cruzada. Haciendo clic sobre el nombre de la técnica deseada ésta queda seleccionada para la generación de los ficheros. Las técnicas disponibles en la aplicación son las siguientes:

- **Validación cruzada k-fold (KFCV):** Este método implica la partición de los datos en  $k$  conjuntos separados, donde  $k$  es el número de conjuntos elegidos. Esta separación se suele realizar generalmente de forma aleatoria en  $k$  disjuntos de un tamaño similar. El conjunto de ensayos de entrenamiento se utiliza para entrenar la red y se puede dividir según el experimento a realizar en entrenamiento y validación.
- **Validación cruzada leave-one-out (LOOCV):** Este método es similar a KFCV pero sólo utiliza un dato para el conjunto de test. Es igual al método KFCV cuando  $k$  es igual al tamaño del conjunto de datos, excepto en que consume más recursos que el anterior, ya que necesita utilizar todos los patrones como conjunto de test y se ejecuta tantas veces como patrones tenga el conjunto original,  $n$ .
- **Remuestreo bootstrap (RBP):** A partir del conjunto original de  $n$  elementos se seleccionan aleatoriamente  $b$  elementos con repetición para producir un nuevo subconjunto de tamaño  $n$ . Este proceso se repite varias veces dando lugar varios conjuntos de datos diferentes. Los datos que no han sido destinados al conjunto de entrenamiento forman parte del conjunto de test mientras que todos los que han sido seleccionados  $n$ , con sus repeticiones se utilizan para entrenar la red. Por tanto el número de patrones destinados a test puede variar, pero el resultado es que se consigue un conjunto de datos de mayor tamaño en entrenamiento y validación.

- **Remuestreo bootstrap (RBP fijo):** Variación del método RBP en el que se define explícitamente el número de datos a utilizar para entrenamiento (se pueden duplicar).
- **Variante de KFCV (VKFCV):** Este método comparte con el KFCV la misma técnica de reparto de patrones basado en la selección aleatoria para generar conjuntos mutuamente excluyentes. El conjunto original de patrones es dividido en dos o tres subconjuntos (train, test y CV) en función de las necesidades de la experimentación a realizar. Esta división se realiza de forma aleatoria pero con la característica de no existir repetición entre los elementos que forman, por lo que se trata de conjuntos disjuntos.

En el desarrollo de esta guía mostraremos de forma completa la técnica de división VKFCV.

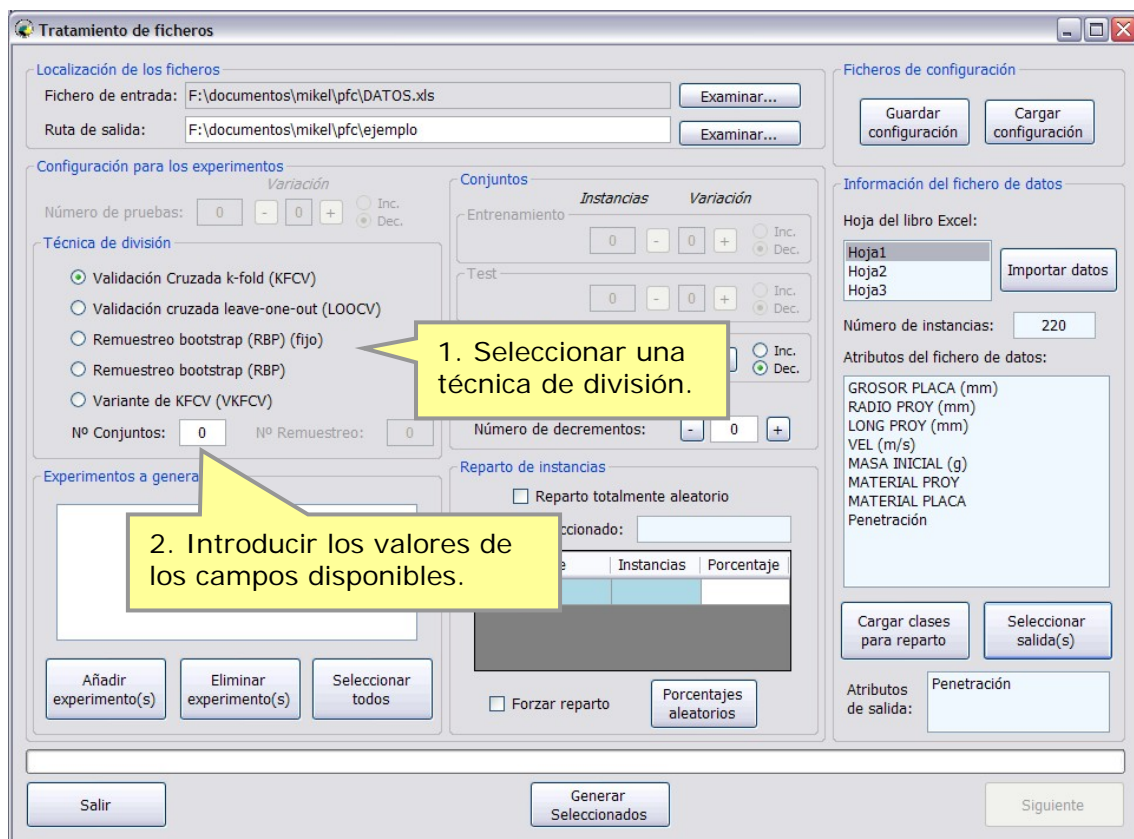
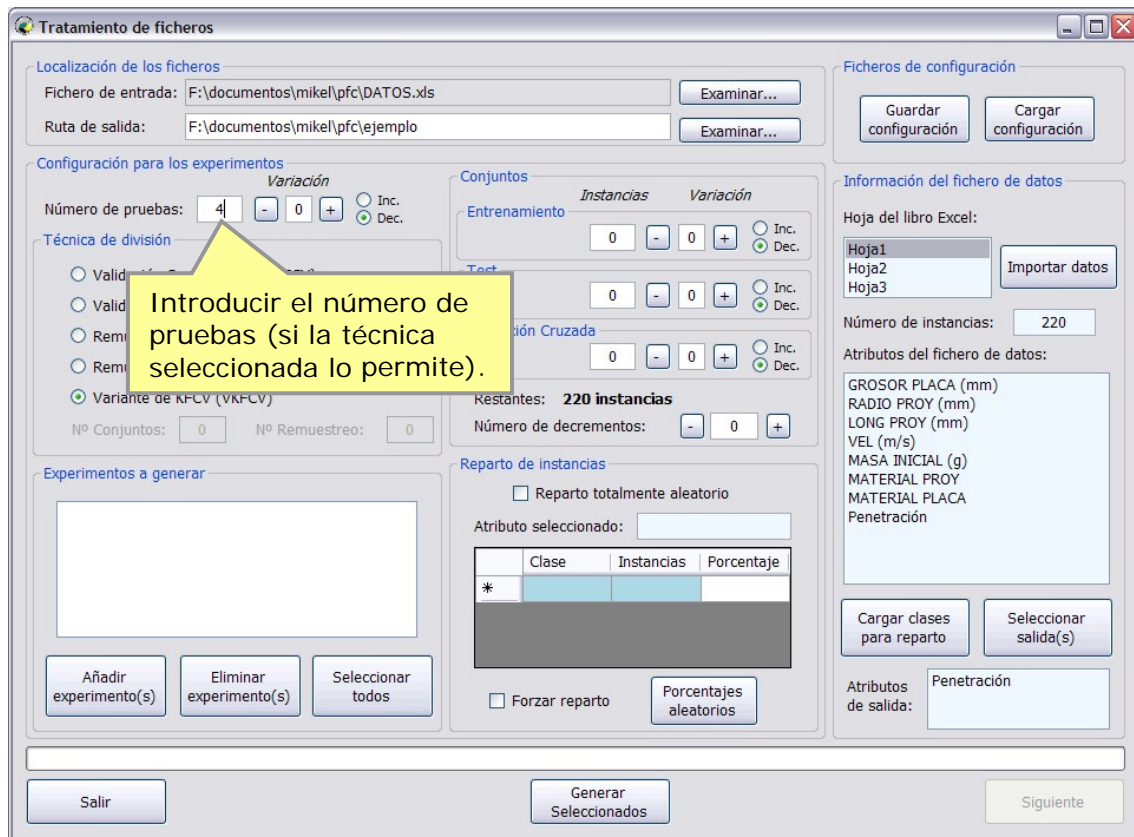


Figura 46. Seleccionar técnica de división



## 11.4.2 Número de pruebas

Una vez seleccionada la técnica de división, la aplicación permite indicar el número de pruebas que se desean generar para el experimento. Para ello el usuario debe introducir el número en el cuadro de texto “Número de pruebas”, tal y como se observa en la siguiente ilustración:



Tratamiento de ficheros

Localización de los ficheros

Fichero de entrada: F:\documentos\mikel\pfc\DATOS.xls Examinar...

Ruta de salida: F:\documentos\mikel\pfc\ejemplo Examinar...

Ficheros de configuración

Guardar configuración Cargar configuración

Configuración para los experimentos

Número de pruebas: 4 - 0 + Inc. Dec.

Técnica de división

☐ Validación  
☐ Validación  
☐ Remuestreo  
☐ Remuestreo  
☒ Variante de KFCV (VKFCV)

Nº Conjuntos: 0 Nº Remuestreo: 0

Experimentos a generar

Añadir experimento(s) Eliminar experimento(s) Seleccionar todos

Conjuntos

Entrenamiento 0 - 0 + Inc. Dec.

Test 0 - 0 + Inc. Dec.

Validación Cruzada 0 - 0 + Inc. Dec.

Restantes: 220 instancias

Número de decrementos: - 0 +

Reparto de instancias

☐ Reparto totalmente aleatorio

Atributo seleccionado: \*

Clase	Instancias	Porcentaje
*		

☐ Forzar reparto Porcentajes aleatorios

Información del fichero de datos

Hoja del libro Excel: Hoja1 Hoja2 Hoja3 Importar datos

Número de instancias: 220

Atributos del fichero de datos:

GROSOR PLACA (mm)  
RADIO PROY (mm)  
LONG PROY (mm)  
VEL (m/s)  
MASA INICIAL (g)  
MATERIAL PROY  
MATERIAL PLACA  
Penetración

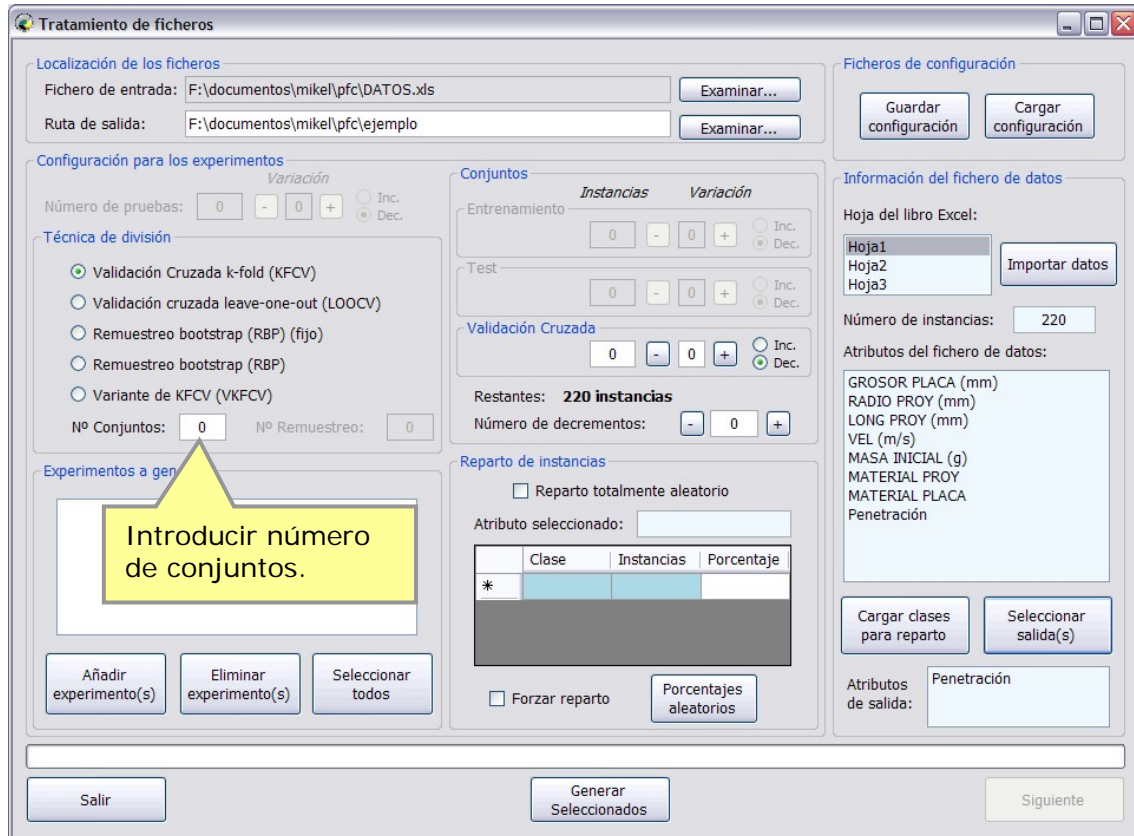
Cargar clases para reparto Seleccionar salida(s)

Atributos de salida: Penetración

Salir Generar Seleccionados Siguiente

Figura 47. Número de pruebas

En caso de haber seleccionado la técnica de división “Validación cruzada k-fold (KFCV)” o “Validación cruzada leave-one-out (LOOCV)”, la aplicación no permite modificar el número de pruebas, ya que éste depende directamente del “Nº Conjuntos” introducido. La siguiente ilustración muestra este aspecto:



**Tratamiento de ficheros**

**Localización de los ficheros**

Fichero de entrada: F:\documentos\mikel\pfc\DATOS.xls [Examinar...]

Ruta de salida: F:\documentos\mikel\pfc\ejemplo [Examinar...]

**Configuración para los experimentos**

Número de pruebas: 0 - 0 + [Inc. Dec.]

**Técnica de división**

- ☒ Validación Cruzada k-fold (KFCV)
- ☐ Validación cruzada leave-one-out (LOOCV)
- ☐ Remuestreo bootstrap (RBP) (fijo)
- ☐ Remuestreo bootstrap (RBP)
- ☐ Variante de KFCV (VKFCV)

Nº Conjuntos: 0 Nº Remuestreo: 0

**Conjuntos**

Entrenamiento: 0 - 0 + [Inc. Dec.]

Test: 0 - 0 + [Inc. Dec.]

**Validación Cruzada**

Restantes: 220 instancias

Número de decrementos: - 0 +

**Reparto de instancias**

☐ Reparto totalmente aleatorio

Atributo seleccionado: [ ]

	Clase	Instancias	Porcentaje
*			

☐ Forzar reparto [Porcentajes aleatorios]

**Ficheros de configuración**

[Guardar configuración] [Cargar configuración]

**Información del fichero de datos**

Hoja del libro Excel: Hoja1 Hoja2 Hoja3 [Importar datos]

Número de instancias: 220

Atributos del fichero de datos:

GROSOR PLACA (mm)  
RADIO PROY (mm)  
LONG PROY (mm)  
VEL (m/s)  
MASA INICIAL (g)  
MATERIAL PROY  
MATERIAL PLACA  
Penetración

[Cargar clases para reparto] [Seleccionar salida(s)]

Atributos de salida: Penetración

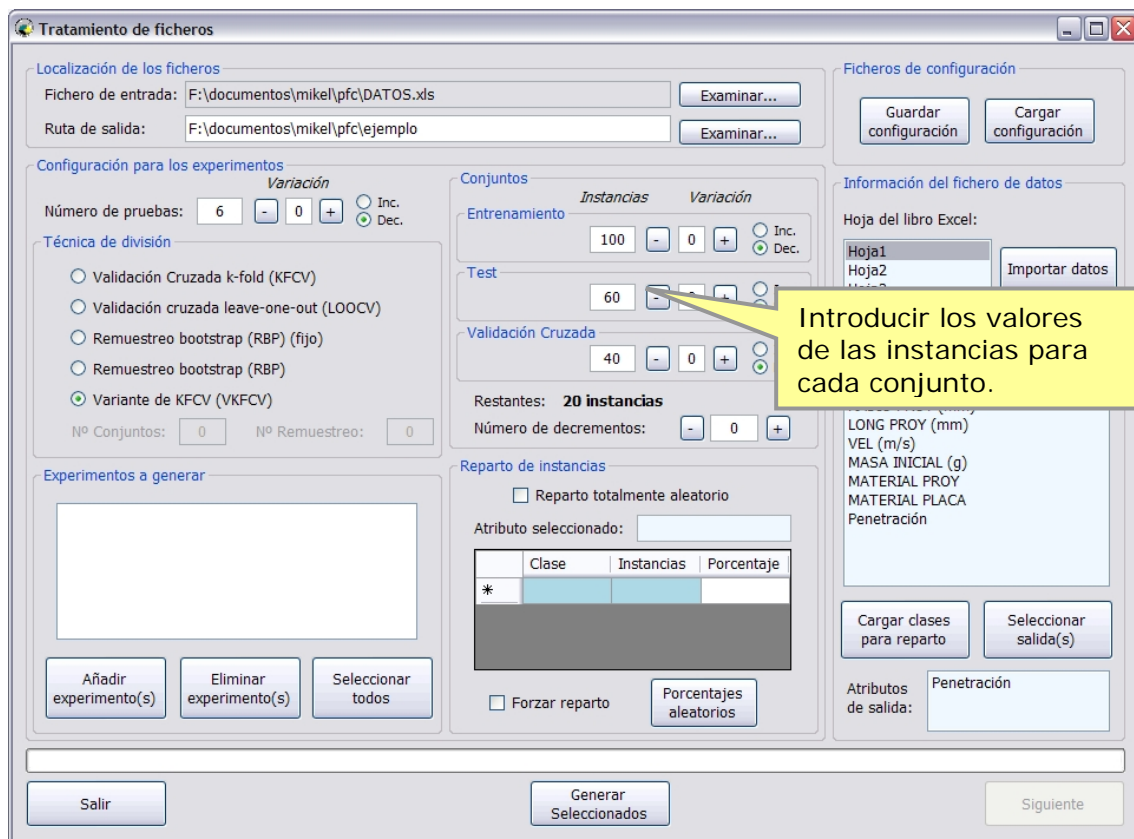
[Salir] [Generar Seleccionados] [Siguiente]

Figura 48. Número de conjuntos

### 11.4.3 Conjuntos de instancias

Para configurar un experimento que se desee preparar la aplicación debe conocer la cantidad de instancias que se desean incluir en cada uno de los conjuntos de instancias. Estos conjuntos son tres: instancias de entrenamiento, instancias de test, e instancias de validación cruzada. En función de la técnica de división seleccionada por el usuario la aplicación permitirá modificar todos los conjuntos de instancias, o únicamente el conjunto de validación cruzada.

La siguiente imagen muestra los cuadros de texto en los que se debe introducir el número de instancias de cada conjunto:



The screenshot shows the 'Tratamiento de ficheros' application window. The 'Conjuntos' section is active, displaying input fields for the number of instances for each set: Entrenamiento (100), Test (60), and Validación Cruzada (40). A yellow callout box points to the 'Test' input field with the text 'Introducir los valores de las instancias para cada conjunto.'

**Localización de los ficheros**

Fichero de entrada: F:\documentos\mikel\pfc\DATOS.xls [Examinar...]  
Ruta de salida: F:\documentos\mikel\pfc\ejemplo [Examinar...]

**Ficheros de configuración**

[Guardar configuración] [Cargar configuración]

**Configuración para los experimentos**

Variación: ☐ Inc. ☒ Dec.

Número de pruebas: 6 [-] [0] [+]

**Técnica de división**

☐ Validación Cruzada k-fold (KFCV)  
☐ Validación cruzada leave-one-out (LOOCV)  
☐ Remuestreo bootstrap (RBP) (fijo)  
☐ Remuestreo bootstrap (RBP)  
☒ Variante de KFCV (VKFCV)

Nº Conjuntos: 0 Nº Remuestreo: 0

**Experimentos a generar**

[Añadir experimento(s)] [Eliminar experimento(s)] [Seleccionar todos]

**Conjuntos**

Instancias Variación

Entrenamiento: 100 [-] [0] [+] ☒ Inc. ☐ Dec.

Test: 60 [-] [0] [+] ☒ Inc. ☐ Dec.

Validación Cruzada: 40 [-] [0] [+] ☒ Inc. ☐ Dec.

Restantes: 20 instancias

Número de decrementos: [-] [0] [+]

**Reparto de instancias**

☐ Reparto totalmente aleatorio

Atributo seleccionado: [ ]

	Clase	Instancias	Porcentaje
*			

☐ Forzar reparto [Porcentajes aleatorios]

**Información del fichero de datos**

Hoja del libro Excel: [Hoja1] [Hoja2] [Importar datos]

LONG PROJ (mm)  
VEL (m/s)  
MASA INICIAL (g)  
MATERIAL PROJ  
MATERIAL PLACA  
Penetración

[Cargar clases para reparto] [Seleccionar salida(s)]

Atributos de salida: [Penetración]

[Salir] [Generar Seleccionados] [Siguiente]

Figura 49. Conjuntos de instancias

### 11.4.4 Reparto de instancias

La aplicación permite que el reparto de instancias entre los distintos conjuntos se realice de dos formas: aleatoria o “dirigida”, según el valor que tome alguno de los atributos seleccionados previamente.

Si se desea que el reparto de instancias sea completamente aleatorio el usuario debe seleccionar la opción “Reparto totalmente aleatorio”, tal y como muestra la siguiente imagen:

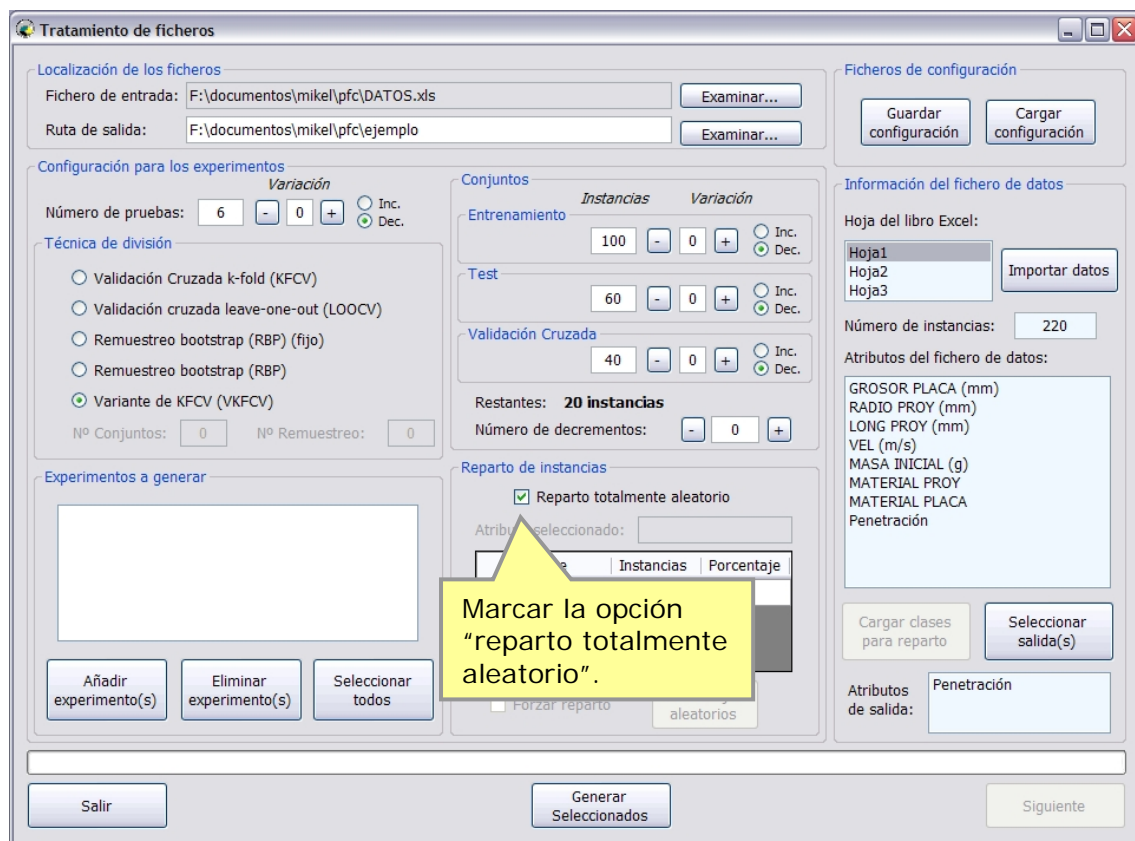


Figura 50. Reparto de instancias totalmente aleatorio

Para utilizar la opción de reparto “dirigido”, en primer lugar se debe seleccionar uno de los atributos y hacer clic sobre el botón “Cargar clases para reparto”. De este modo la aplicación cargará todos los posibles valores que encuentre para ese atributo en la tabla de “Reparto de instancias”. En dicha tabla el usuario deberá indicar los porcentajes que desea incluir de cada clase en cada conjunto de instancias, o bien hacer clic sobre el botón “Porcentajes aleatorios” para que la aplicación asigne los porcentajes de manera aleatoria

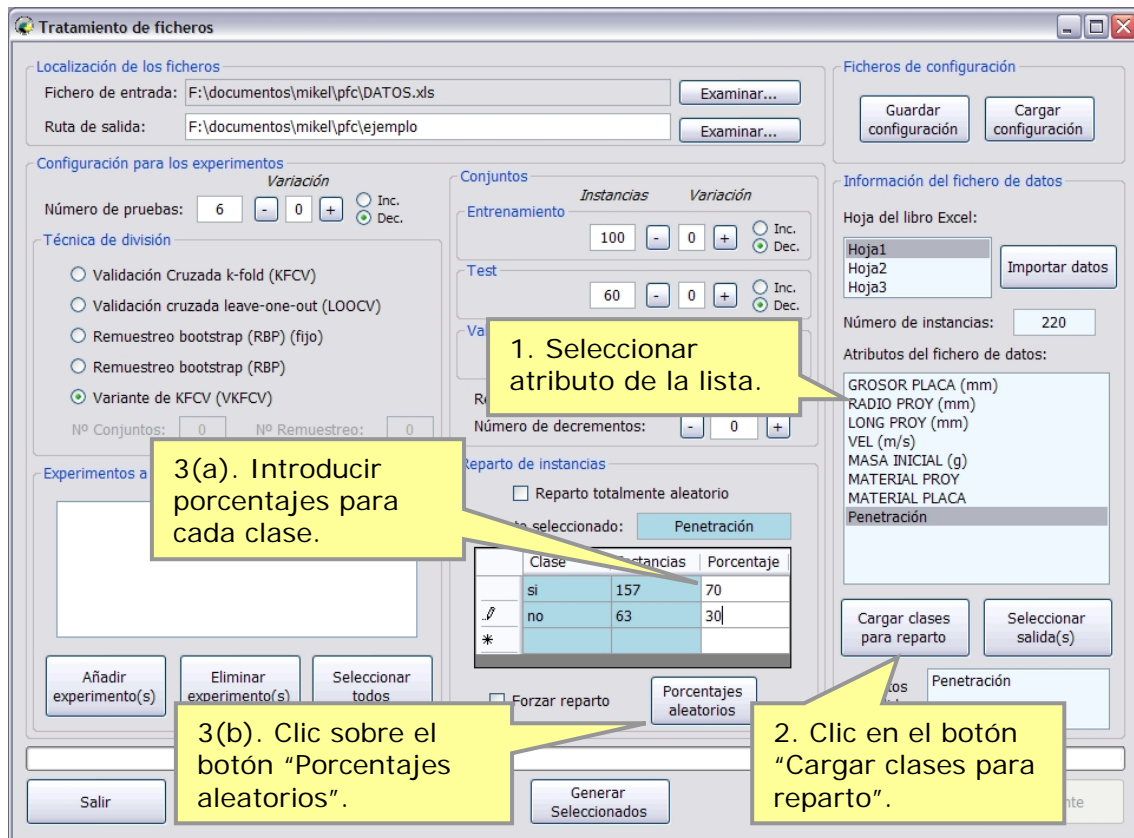


Figura 51. Reparto de instancias dirigido

La opción "Forzar reparto" da la posibilidad de completar la cantidad de instancias del conjunto con instancias de otra clase si la cantidad de instancias de la clase deseada no son suficientes. En caso de no seleccionar esta opción la aplicación mostrará un aviso si se ha tenido que modificar alguna cantidad de instancias durante el proceso.

### 11.4.5 Añadir experimentos a preparar

Una vez definidos todos los parámetros de configuración para el experimento se hace clic sobre el botón “Añadir experimento(s)” y este se añade a la lista de “Experimentos a generar”, tal y como se aprecia en la siguiente ilustración:

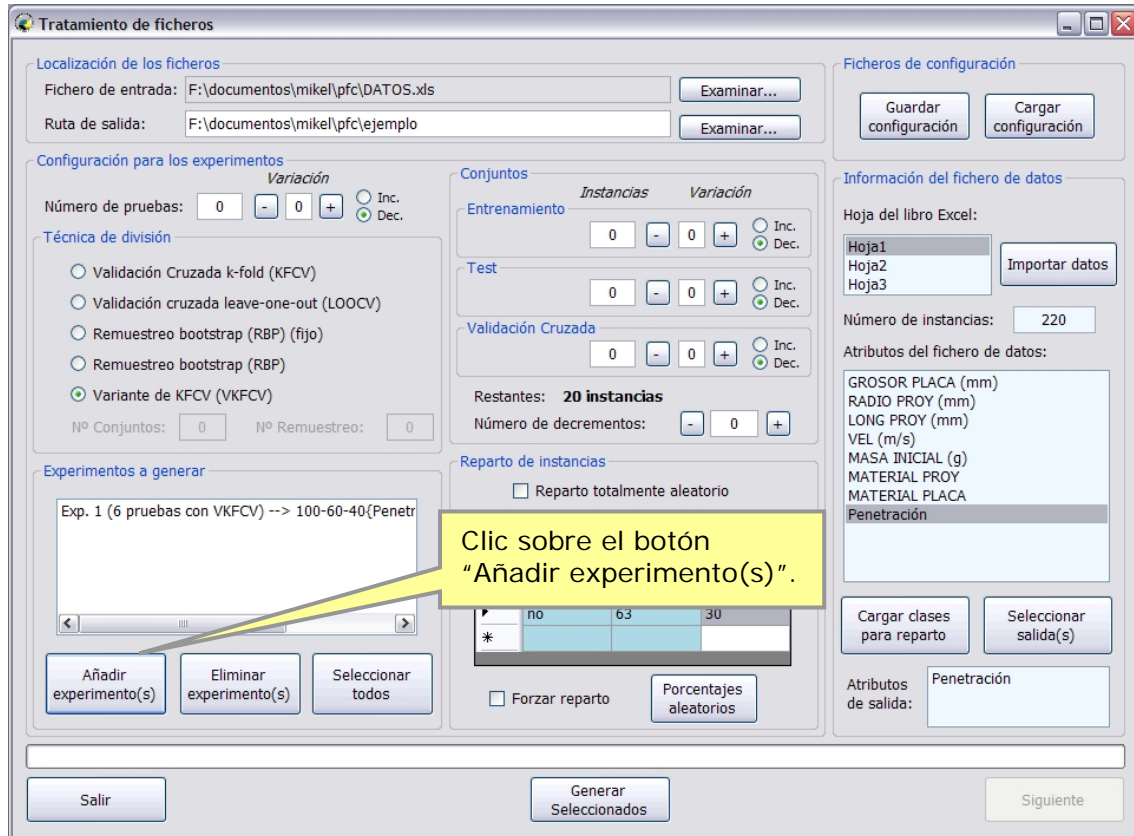


Figura 52. Añadir experimento

El experimento se añade a la lista de “Experimentos a preparar” con el nombre Experimento y un número secuencial correspondiente a su lugar en la lista. Además, se almacenan los parámetros definidos previamente de forma abreviada (número de pruebas, número de ensayos para Train, número de ensayos para Test, número de ensayos para Validación Cruzada, atributo de reparto nombres y porcentajes de cada clase).



## 11.4.6 Añadir experimentos de forma automatizada

Para simplificar la función de añadir múltiples experimentos de forma automatizada, la aplicación permite, una vez definidos los parámetros del experimento, indicar el decremento a realizar sobre cada uno de ellos junto con el número de decrementos a realizar. En la siguiente figura se observa un ejemplo:

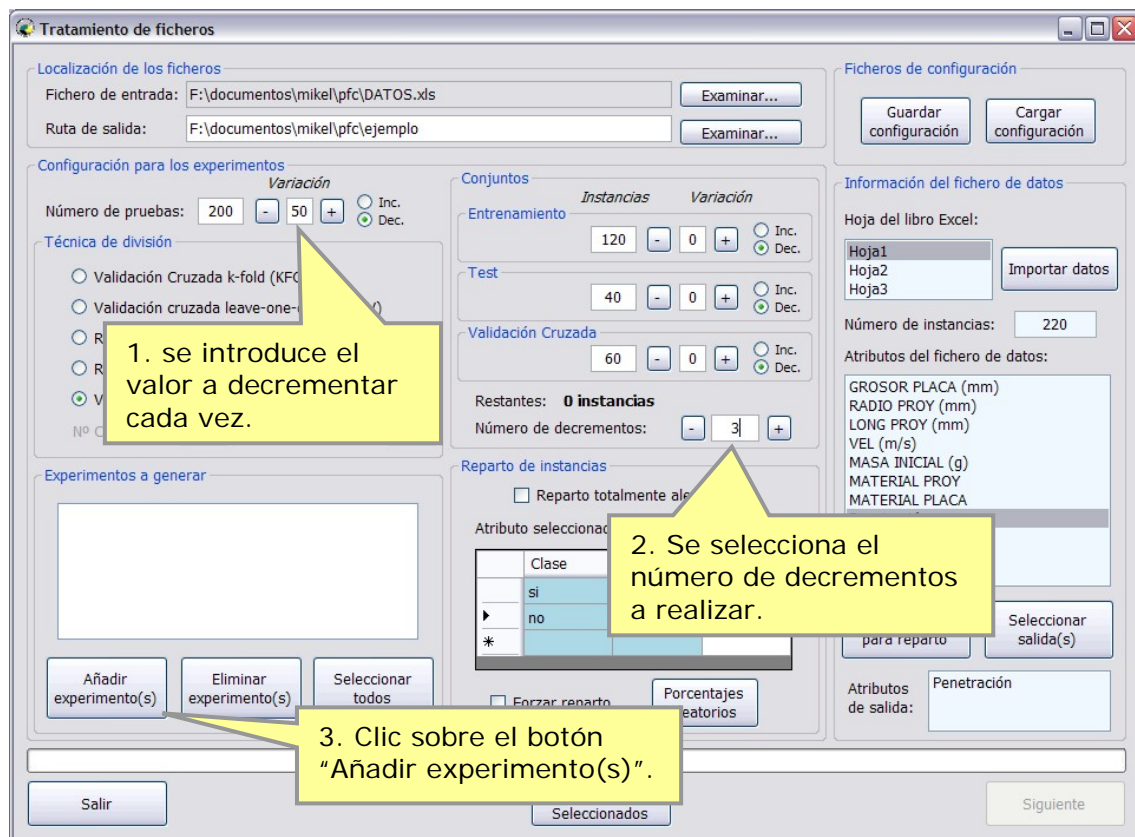
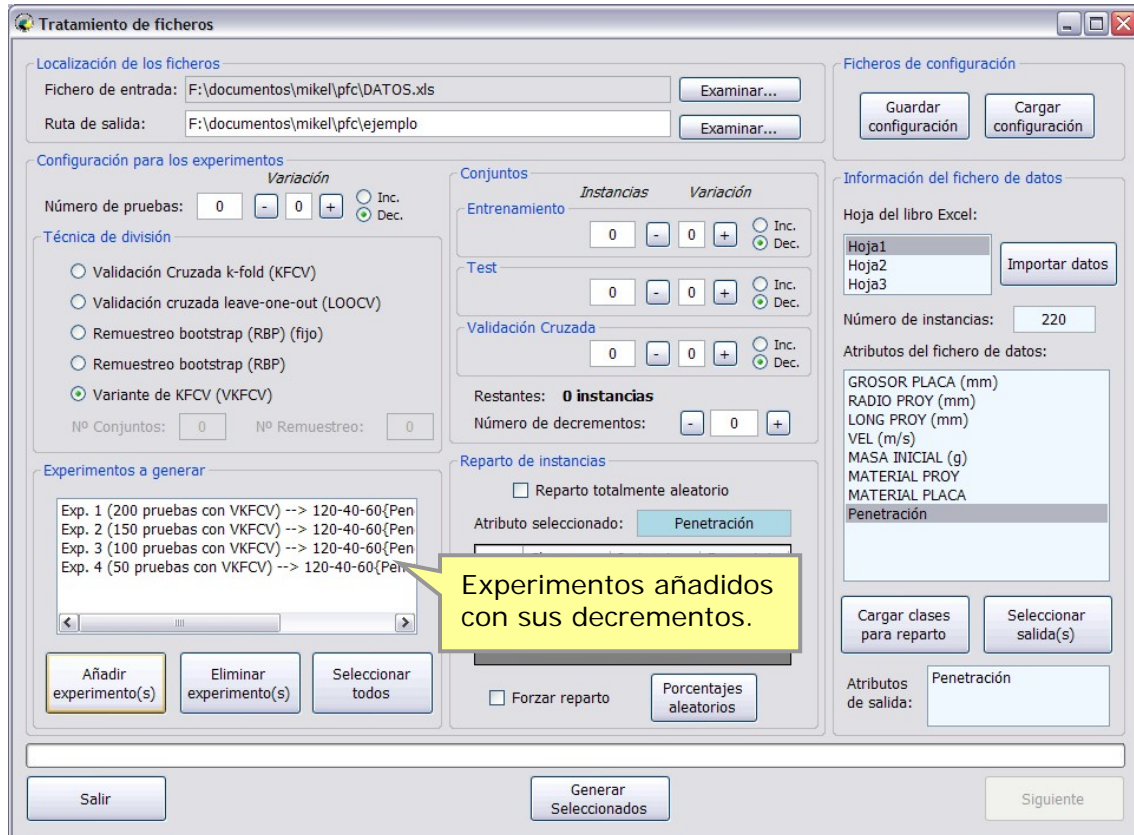


Figura 53. Añadir experimento de forma automatizada

En este caso se añadirán 4 experimentos a preparar a la lista una vez se haga clic en el botón "Añadir Experimento(s)", el experimento con los parámetros definidos junto con 3 decrementos del mismo, variando el número de pruebas (de 50 en 50 unidades).

En la siguiente figura podemos observar los experimentos a preparar después de hacer clic en el botón “añadir Experimento(s)”:



The screenshot shows the 'Tratamiento de ficheros' (File Processing) window. The 'Localización de los ficheros' (File Location) section has 'Fichero de entrada' (Input file) set to 'F:\documentos\mikel\pfc\DATOS.xls' and 'Ruta de salida' (Output path) set to 'F:\documentos\mikel\pfc\ejemplo'. The 'Configuración para los experimentos' (Experiment Configuration) section includes 'Variación' (Variation) with 'Número de pruebas' (Number of tests) set to 0, and 'Técnica de división' (Division technique) with 'Validación Cruzada k-fold (KFCV)' selected. The 'Conjuntos' (Sets) section shows 'Entrenamiento' (Training) and 'Test' with 'Número de decrementos' (Number of decrements) set to 0. The 'Reparto de instancias' (Instance distribution) section has 'Reparto totalmente aleatorio' (Totally random distribution) checked. The 'Experimentos a generar' (Experiments to generate) section lists four experiments: 'Exp. 1 (200 pruebas con VKFCV) --> 120-40-60{Pen}', 'Exp. 2 (150 pruebas con VKFCV) --> 120-40-60{Pen}', 'Exp. 3 (100 pruebas con VKFCV) --> 120-40-60{Pen}', and 'Exp. 4 (50 pruebas con VKFCV) --> 120-40-60{Pen}'. A yellow callout box points to the list of experiments with the text 'Experimentos añadidos con sus decrementos.' (Experiments added with their decrements). The 'Ficheros de configuración' (Configuration files) section has 'Guardar configuración' (Save configuration) and 'Cargar configuración' (Load configuration) buttons. The 'Información del fichero de datos' (Data file information) section shows 'Hoja del libro Excel' (Excel workbook sheet) set to 'Hoja1' and 'Número de instancias' (Number of instances) set to 220. The 'Atributos del fichero de datos' (Data file attributes) section lists attributes: 'GROSOR PLACA (mm)', 'RADIO PROY (mm)', 'LONG PROY (mm)', 'VEL (m/s)', 'MASA INICIAL (g)', 'MATERIAL PROY', 'MATERIAL PLACA', and 'Penetración'. The 'Atributos de salida' (Output attributes) section has 'Penetración' selected. The bottom of the window has 'Salir' (Exit), 'Generar Seleccionados' (Generate Selected), and 'Siguiente' (Next) buttons.

Figura 54. Resultado añadir experimentos de forma automatizada



## 11.4.7 Eliminar experimentos

Una vez añadido un experimento a la lista (o varios), la aplicación permite eliminarlos simplemente seleccionándolos en la lista y haciendo clic sobre el botón “Eliminar experimento(s)”. (NOTA: Para seleccionar varios usar CONTROL + CLIC).

En el siguiente ejemplo se quieren eliminar los experimentos 1 y 4:

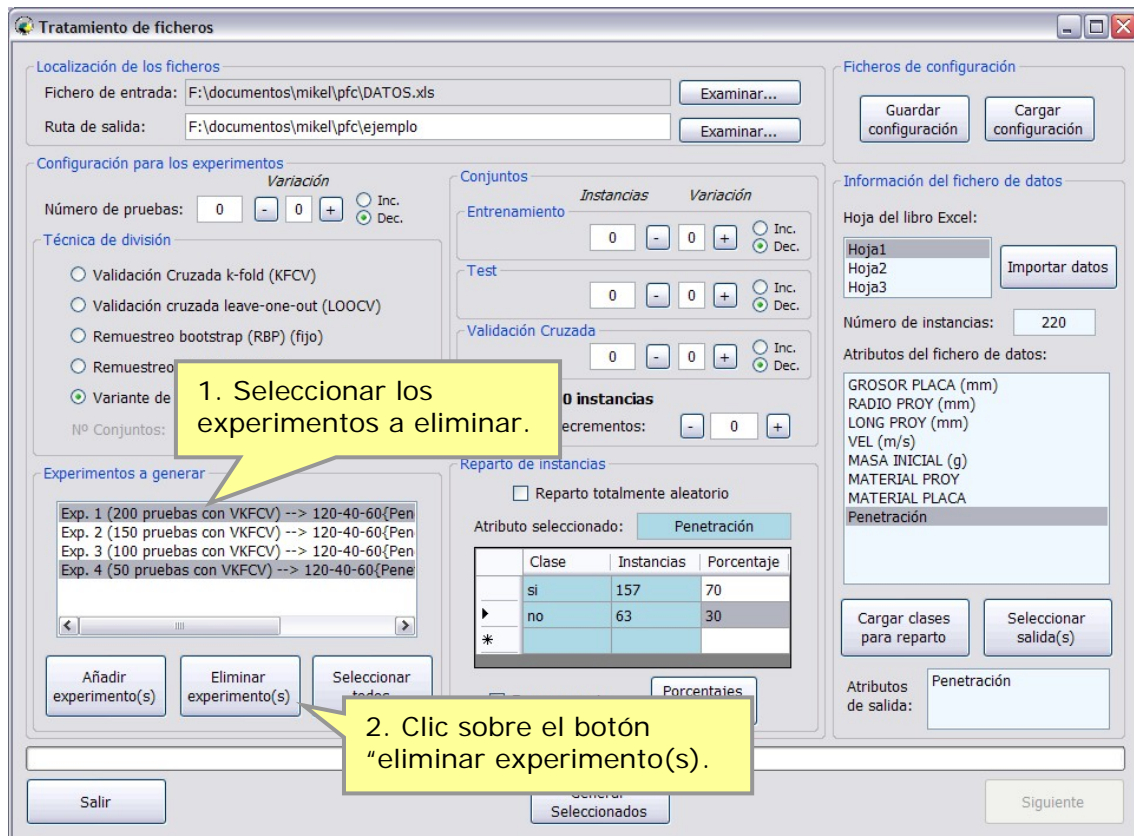


Figura 55. Eliminar Varios Experimentos



En la siguiente figura podemos observar que los experimentos seleccionados han sido eliminados:

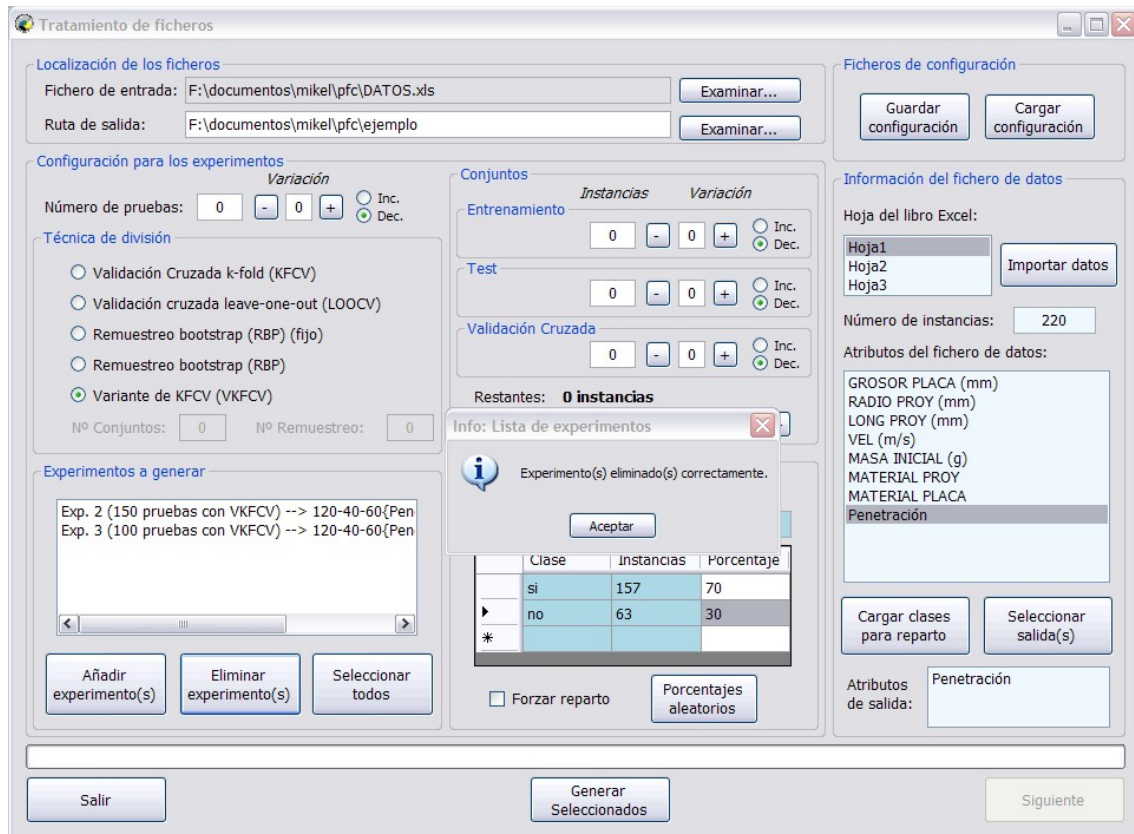


Figura 56. Resultado de experimentos eliminados

## 11.5 Generación de ficheros

Una vez que el usuario ha añadido los experimentos a la lista “Experimentos a generar”, seleccionamos los experimentos que deseamos generar. Para ello podemos seleccionarlos usando CONTROL+CLIC o pulsando el botón “Seleccionar todos” si queremos entrenar todos los experimentos de la lista.

Durante todo el proceso de generación la aplicación muestra una barra de progreso del experimento que se esté procesando, permitiendo al usuario supervisar dicho proceso. A continuación se puede observar el indicador situado en la parte inferior del interfaz:

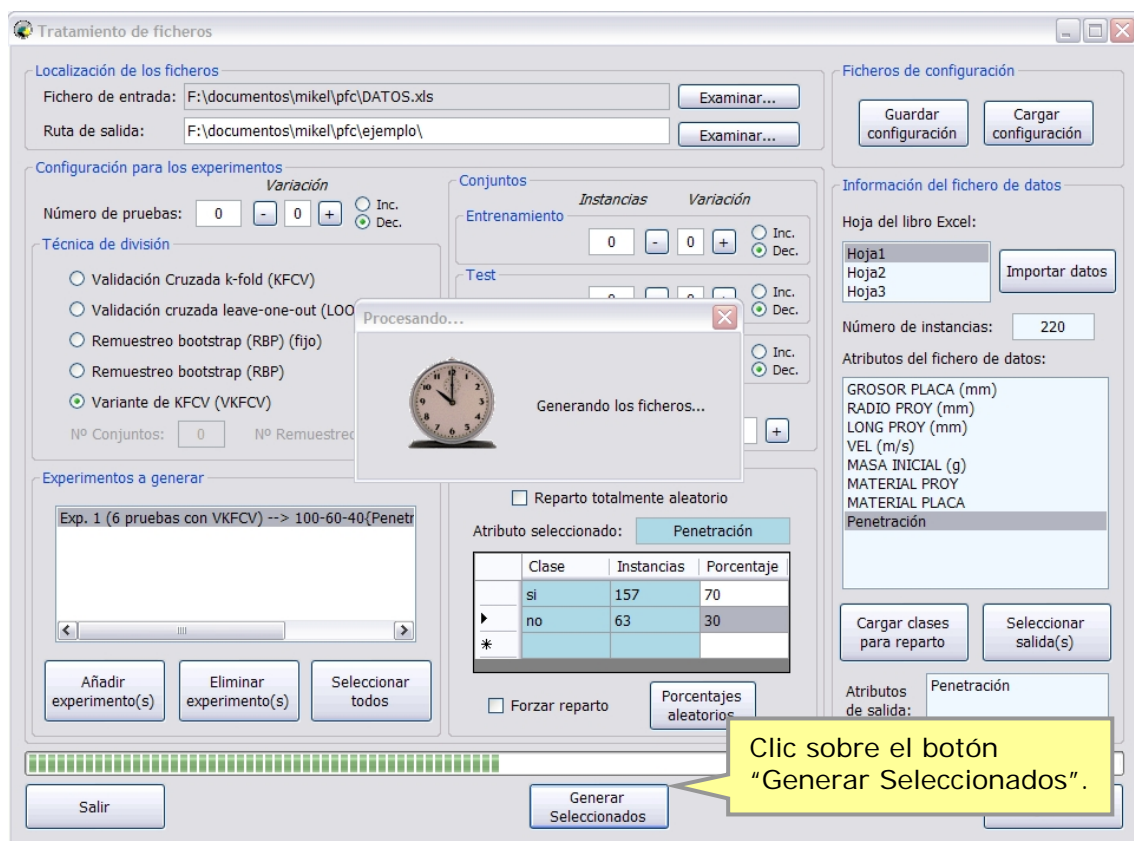


Figura 57. Generación de ficheros

Una vez terminado el proceso de generación, la aplicación muestra un mensaje informado al usuario:

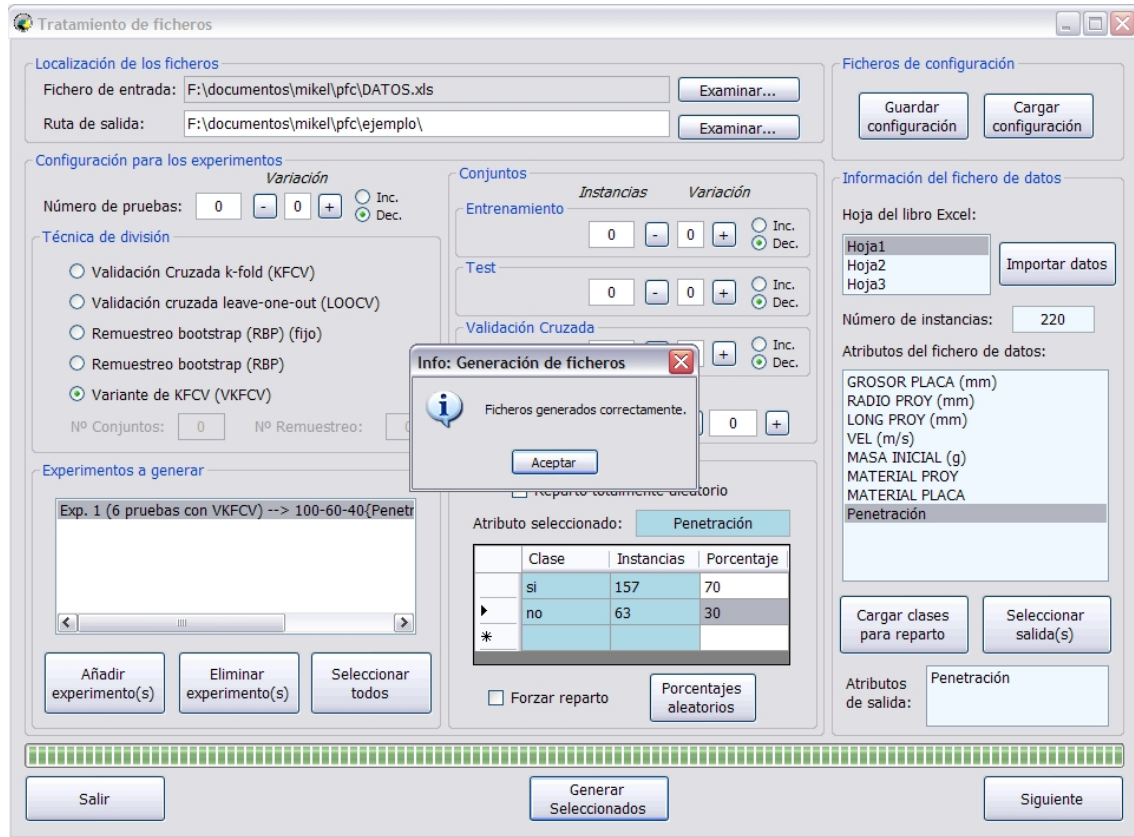


Figura 58. Resultado generación ficheros

El usuario podrá encontrar los ficheros generados en la carpeta indicada en el cuadro de texto "Ruta de salida". La aplicación crea una carpeta independiente para cada uno de los experimentos generados. Dentro de cada carpeta de experimento se puede encontrar tres tipos de ficheros:

- **Ficheros de entrenamiento o train:** Son los ficheros designados para el entrenamiento de la red de neuronas artificiales. Su nomenclatura sigue el siguiente formato: "*TrainXX.txt*", donde XX indica el número de prueba a la que corresponde.
- **Ficheros de validación o test:** Son los ficheros designados para la validación de la red, una vez entrenada. Su nomenclatura es del tipo: "*TestXX.txt*", donde XX indica el número de prueba a la que corresponde.

- **Ficheros de validación cruzada o cross validation:** Son los ficheros designados para el entrenamiento de la red de neuronas artificiales, en caso de utilizar como criterio de parada la validación cruzada. Su nomenclatura es: "ValXX.txt", donde XX indica el número de prueba a la que corresponde. Si el usuario ha configurado el experimento sin este tipo de ficheros, no se encontrará ninguno dentro de la carpeta del mismo.

La siguiente imagen muestra un posible resultado de la generación de los ficheros de un experimento:

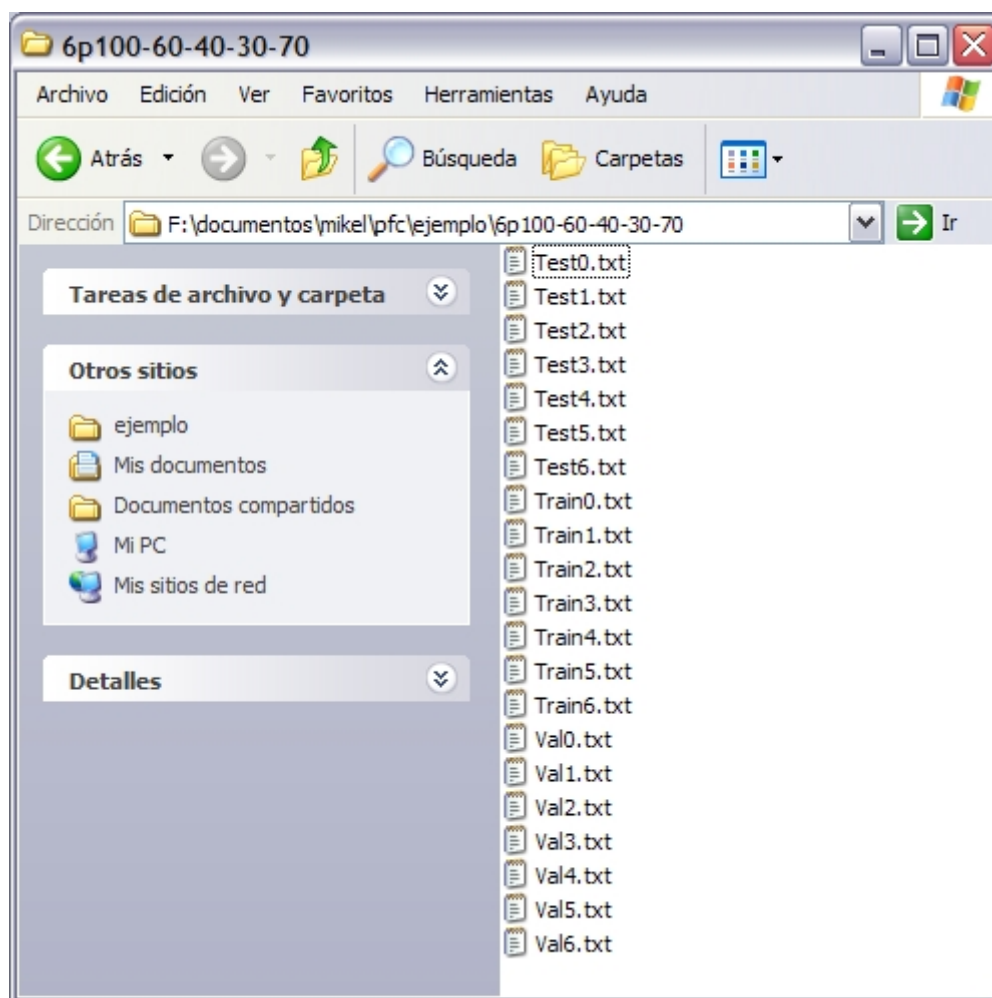


Figura 59. Ficheros generados

## 11.6 Gestión de la configuración

Es posible que el usuario desee almacenar la configuración de datos y experimentos realizada previamente en la aplicación. De este modo podrá realizar la generación de ficheros en otro momento, o bien trabajar con una configuración en cualquier momento que desee. La aplicación permite tanto guardar como cargar un fichero de configuración, tal y como se describe a continuación:

### 11.6.1 Guardar configuración

Cuando se ha completado la configuración de los datos y experimentos, el usuario debe hacer clic sobre el botón “Guardar configuración” situado en la parte superior derecha de la pantalla. La aplicación mostrará un cuadro de diálogo en el que se podrá seleccionar el fichero en el que almacenar la configuración actual. La siguiente imagen muestra un ejemplo de esta operación:

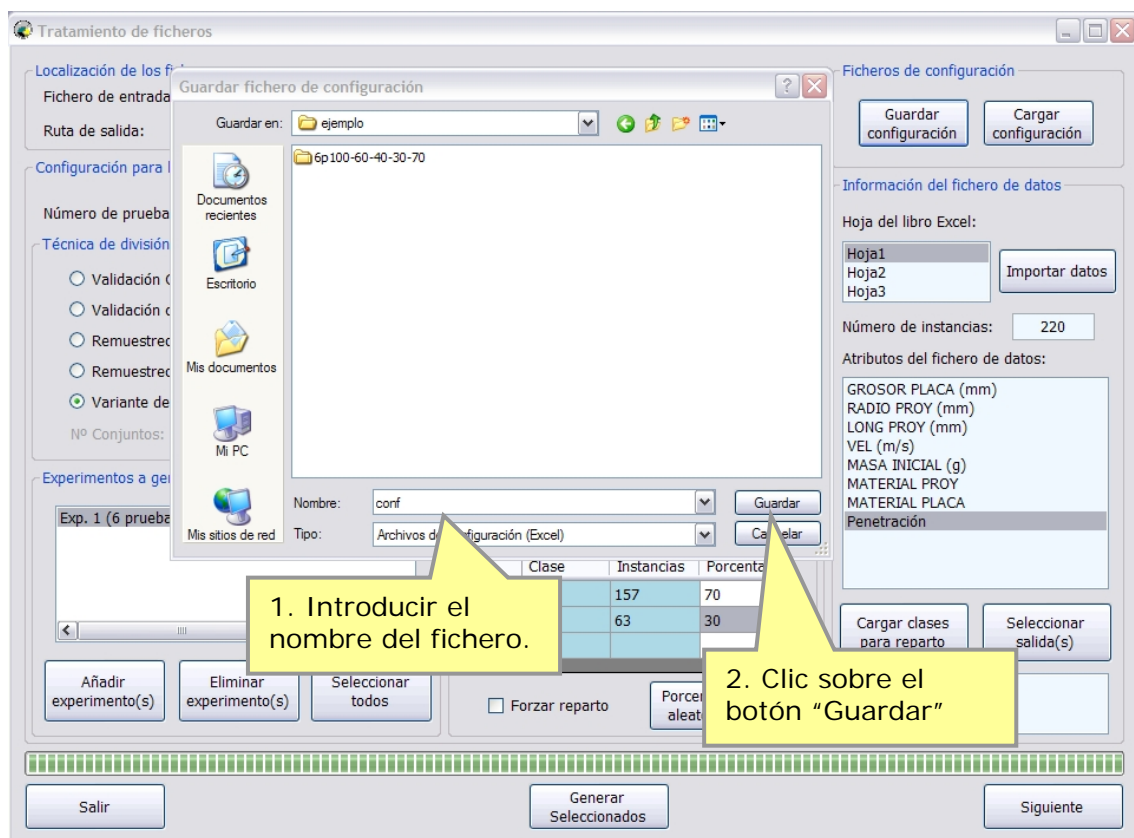


Figura 60. Guardar configuración



Una vez guardada la configuración, la aplicación informará al usuario que la operación se ha realizado satisfactoriamente:

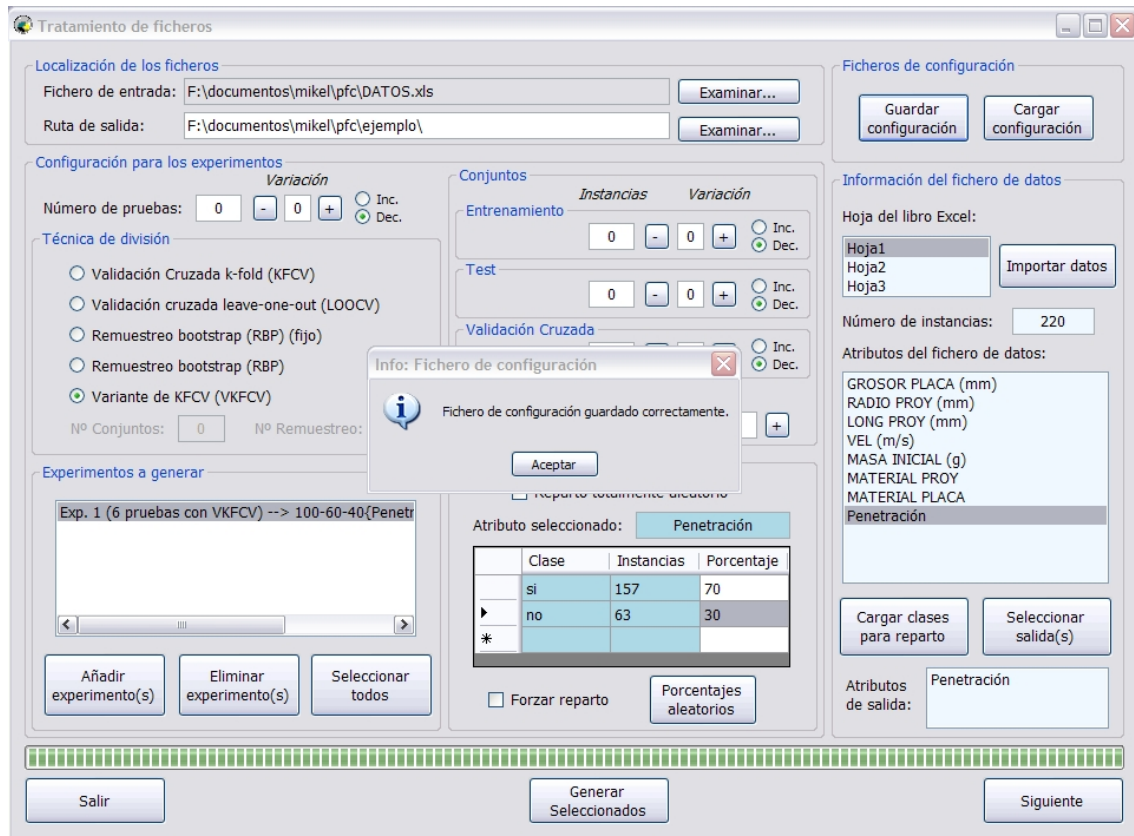


Figura 61. Resultado guardar configuración

## 11.6.2 Cargar configuración

Si previamente se ha guardado la configuración de la aplicación en un fichero, éste podrá ser cargado en cualquier momento, devolviendo la aplicación al estado de configuración en el que se encontraba en el momento de guardar el fichero en cuestión. Para ello se debe hacer clic sobre el botón “Cargar configuración” y seleccionar el archivo de configuración en el cuadro de diálogo que muestra la aplicación. A continuación se puede observar el funcionamiento:

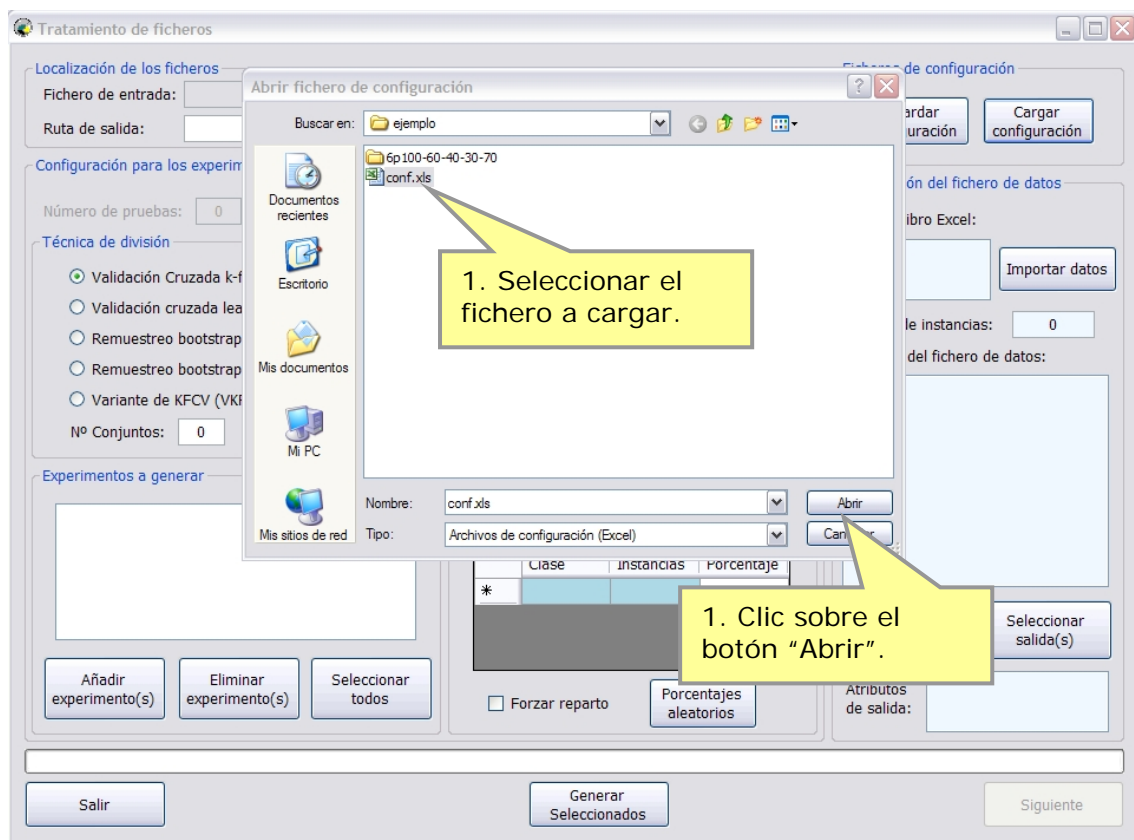


Figura 62. Cargar configuración



Una vez cargada la configuración, la aplicación informará al usuario que la operación se ha realizado satisfactoriamente:

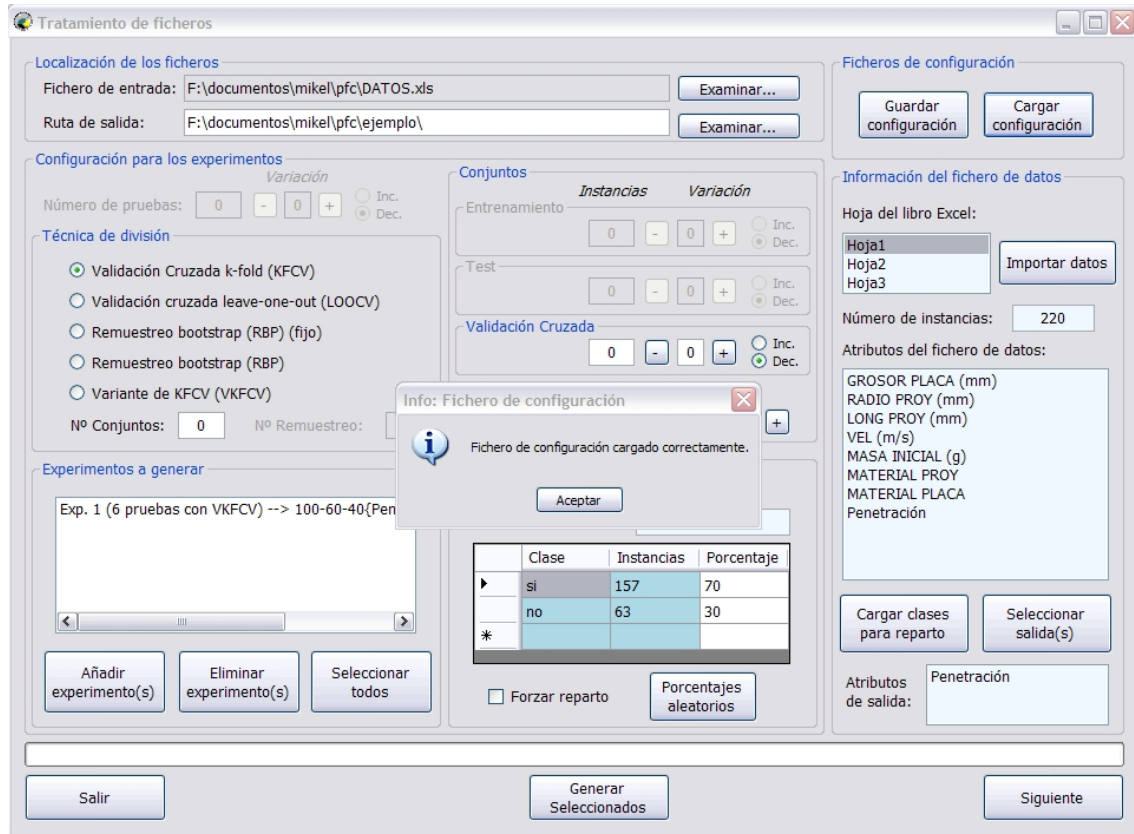
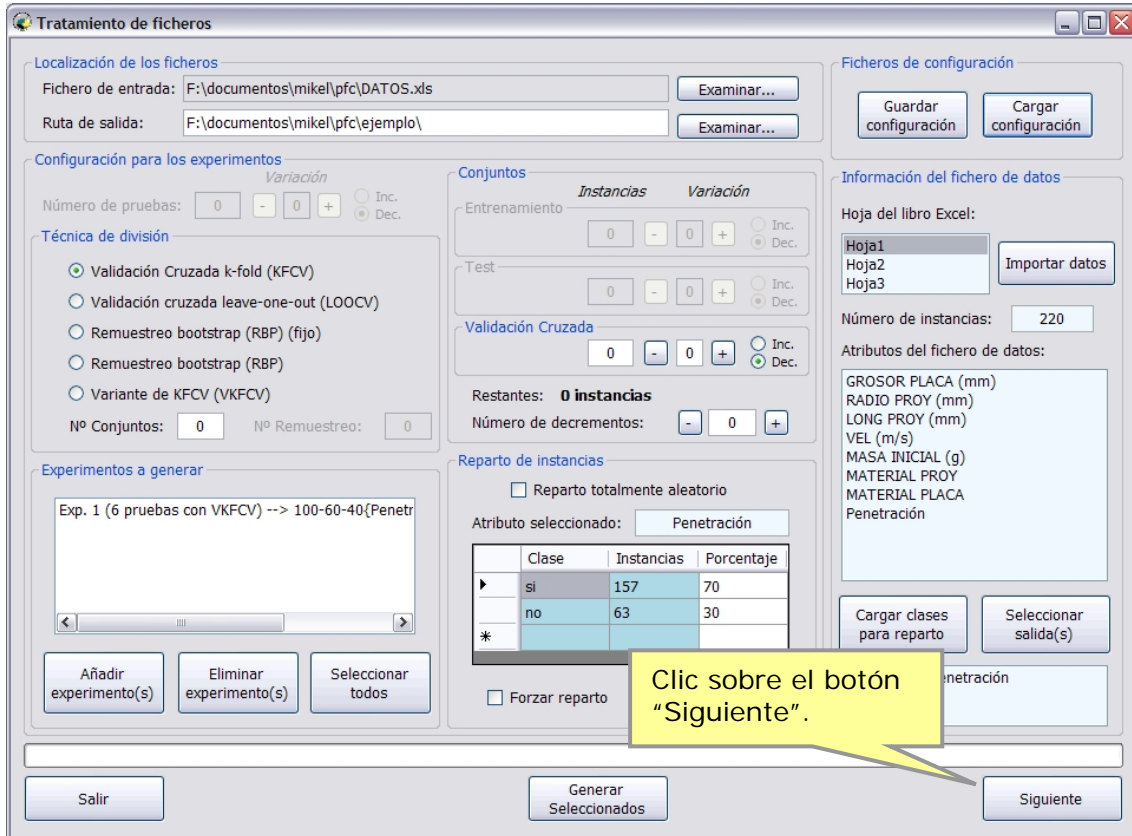


Figura 63. Resultado cargar configuración

## 11.7 Interfaz entrenamiento

Una vez generados los ficheros haremos clic en el botón “Siguiente para poder visualizar la interfaz de entrenamiento:



**Localización de los ficheros**

Fichero de entrada: F:\documentos\mikel\pfc\DATOS.xls Examinar...

Ruta de salida: F:\documentos\mikel\pfc\ejemplo\ Examinar...

**Configuración para los experimentos**

Número de pruebas: 0 - + Inc. Dec.

**Técnica de división**

- ☒ Validación Cruzada k-fold (KFCV)
- ☐ Validación cruzada leave-one-out (LOOCV)
- ☐ Remuestreo bootstrap (RBP) (fijo)
- ☐ Remuestreo bootstrap (RBP)
- ☐ Variante de KFCV (VKFCV)

Nº Conjuntos: 0 Nº Remuestreo: 0

**Experimentos a generar**

Exp. 1 (6 pruebas con VKFCV) --> 100-60-40(Penetración)

Añadir experimento(s) Eliminar experimento(s) Seleccionar todos

**Conjuntos**

**Entrenamiento** 0 - + Inc. Dec.

**Test** 0 - + Inc. Dec.

**Validación Cruzada** 0 - + Inc. Dec.

Restantes: 0 instancias

Número de decrementos: - 0 +

**Reparto de instancias**

☐ Reparto totalmente aleatorio

Atributo seleccionado: Penetración

	Clase	Instancias	Porcentaje
▶	si	157	70
	no	63	30
*			

☐ Forzar reparto

**Ficheros de configuración**

Guardar configuración Cargar configuración

**Información del fichero de datos**

Hoja del libro Excel: Hoja1, Hoja2, Hoja3 Importar datos

Número de instancias: 220

Atributos del fichero de datos: GROSOR PLACA (mm), RADIO PROY (mm), LONG PROY (mm), VEL (m/s), MASA INICIAL (g), MATERIAL PROY, MATERIAL PLACA, Penetración

Cargar clases para reparto Seleccionar salida(s)

Salir Generar Seleccionados Siguiente

Clic sobre el botón “Siguiente”.

Figura 64. Mostrar interfaz entrenamiento

A continuación podemos ver en la imagen la interfaz de entrenamiento de la aplicación:

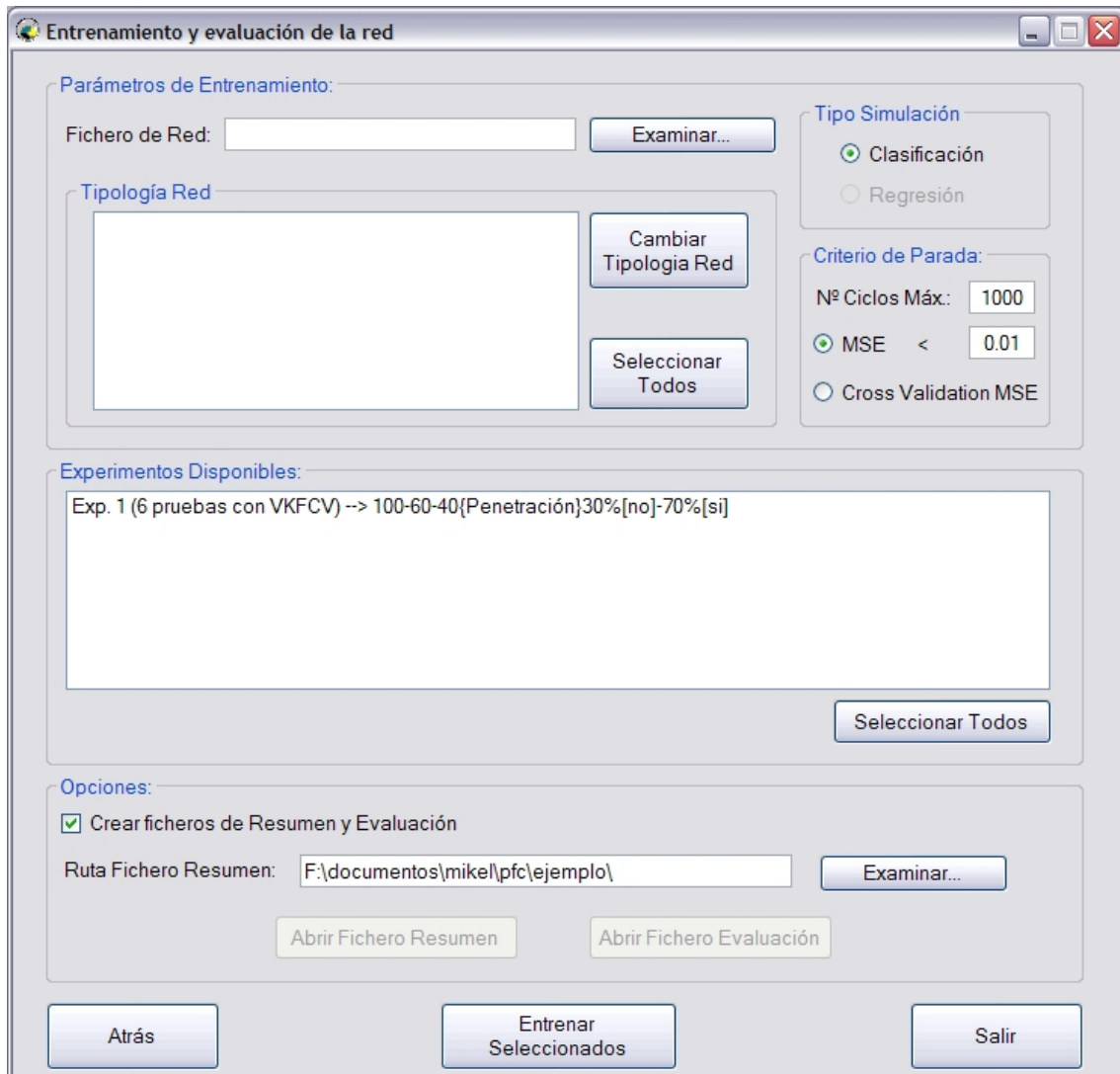


Figura 65. Interfaz de Entrenamiento

NOTA: Pulsando sobre el botón Atrás podemos volver a la interfaz de tratamiento tal y como la habíamos dejado.

## 11.8 Localización ficheros red

Para especificar los ficheros de red, se debe seleccionar la ruta mediante el ratón utilizando el botón "Examinar..." situado junto al cuadro de texto "Fichero de red", a través del cuadro de diálogo que mostrará la aplicación. La siguiente ilustración muestra el proceso:

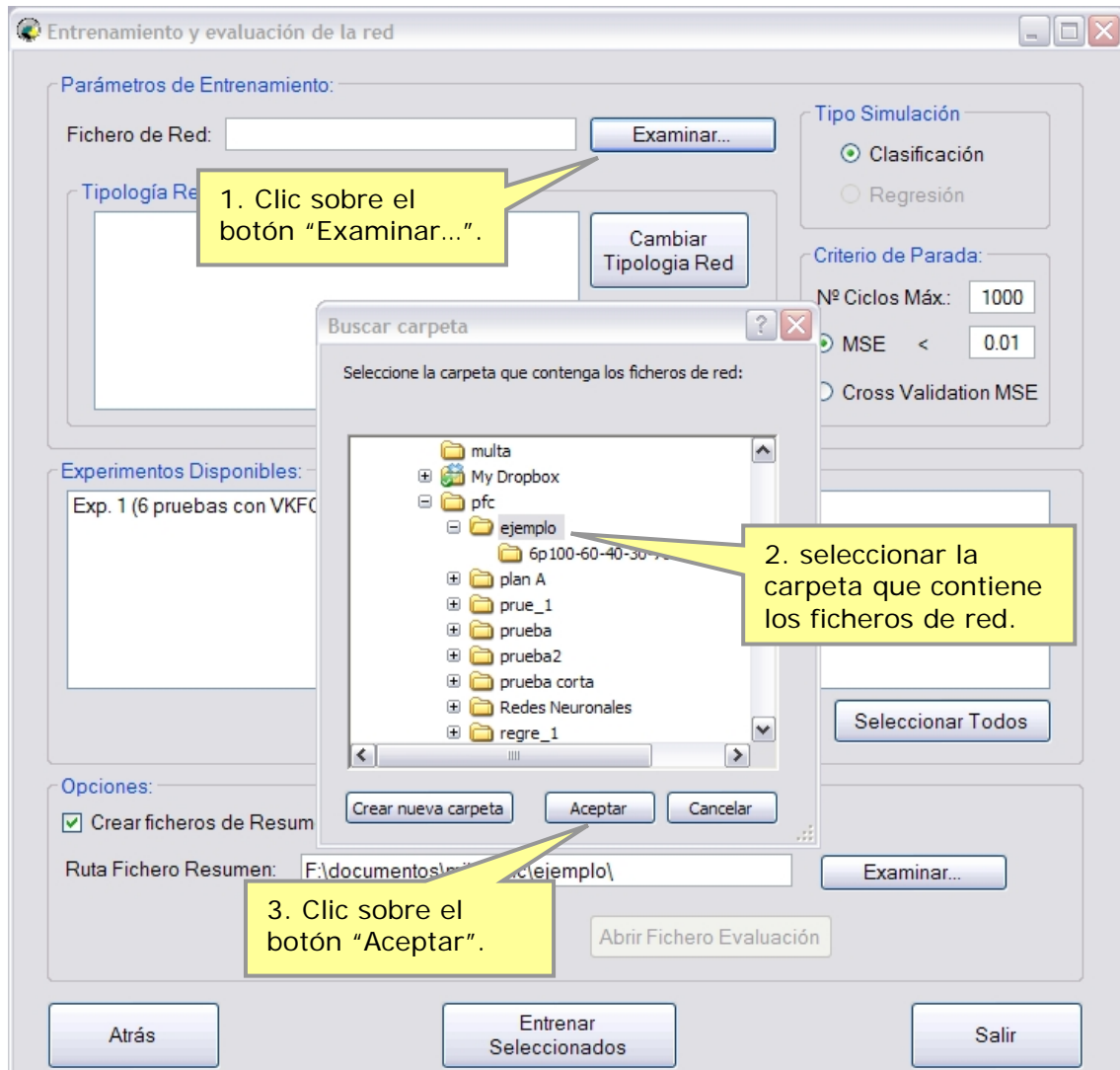


Figura 66. Selección ruta ficheros de red

NOTA: Para crear un fichero de red remítase al punto 14 Anexo c: Creación de una red de neuronas en Neurosolutions.

En la siguiente imagen podemos observar como la aplicación ha identificado la tipología de la red neuronal de cada fichero de red identificado en la ruta especificada:

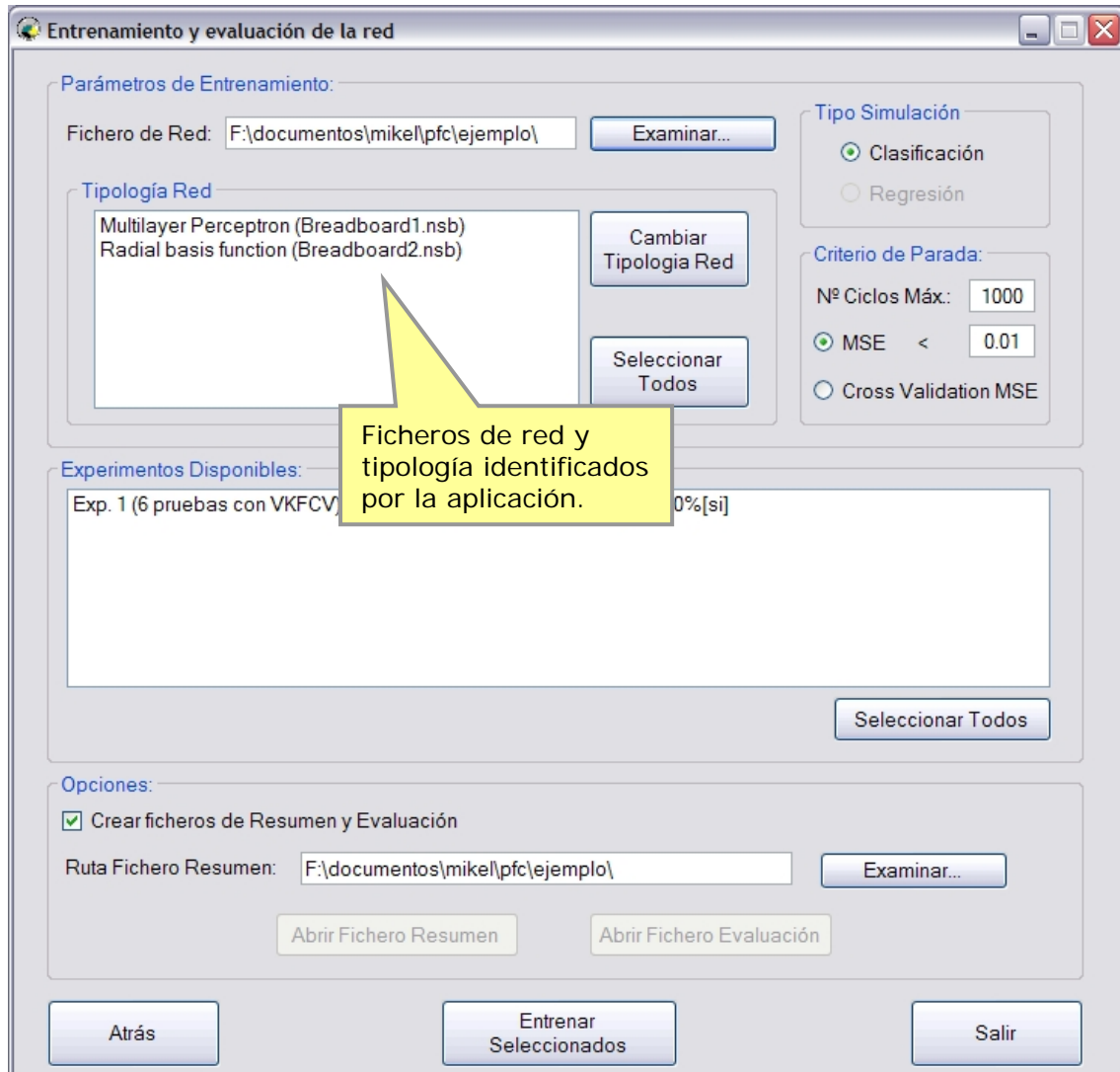


Figura 67. Identificación ficheros de red

## 11.9 Cambiar manualmente la tipología de red

La aplicación permite cambiar manualmente la tipología de una red de neuronas si el usuario lo requiere.

Para ello se debe seleccionar un fichero de red de la lista "Tipología Red" y hacer clic en el botón "Cambiar Tipología Red". El usuario podrá seleccionar la tipología que crea conveniente a través del cuadro de dialogo que mostrara la aplicación. La siguiente ilustración muestra el proceso:

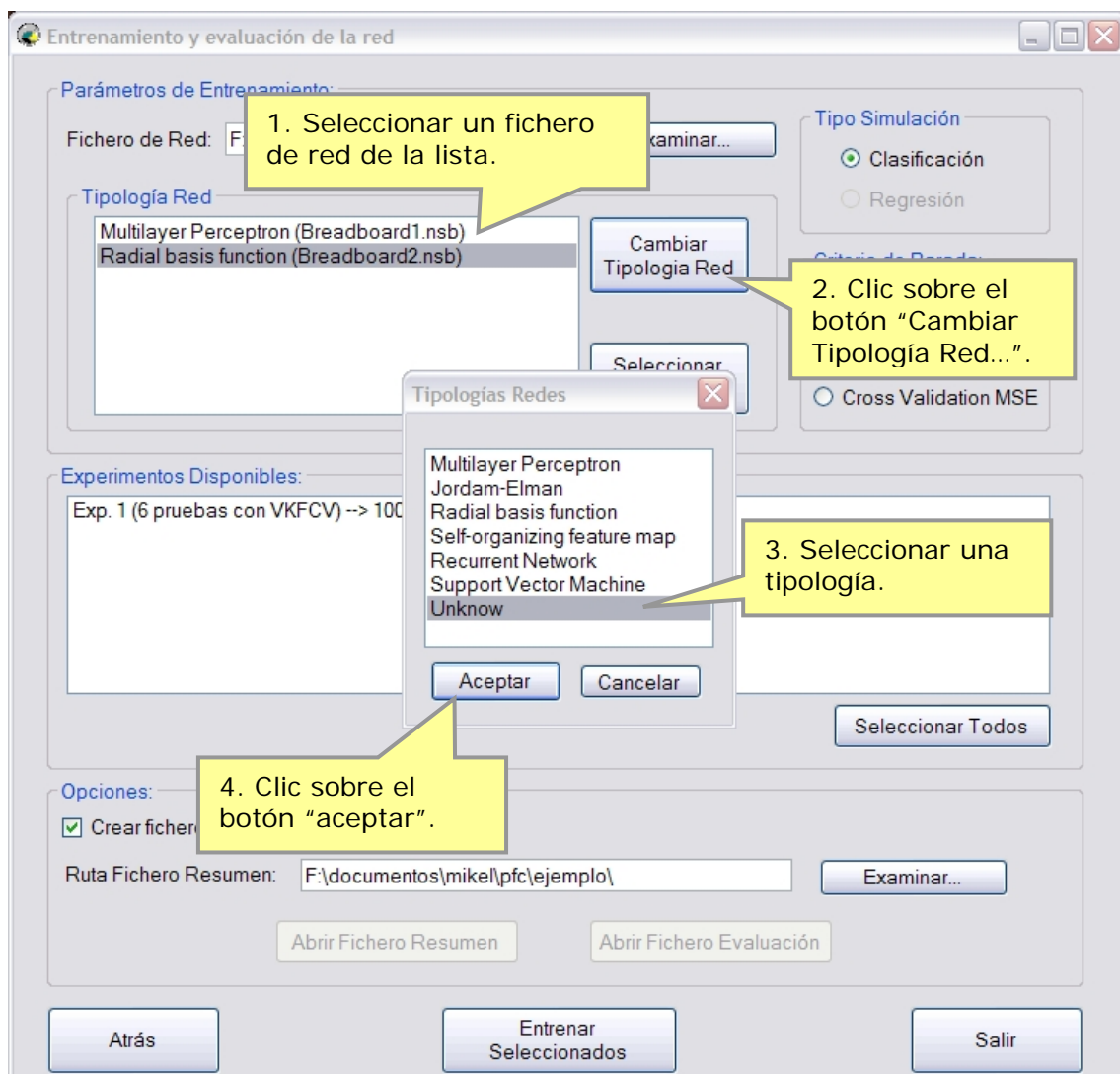


Figura 68. Selección de tipología

La siguiente ilustración muestra el resultado:

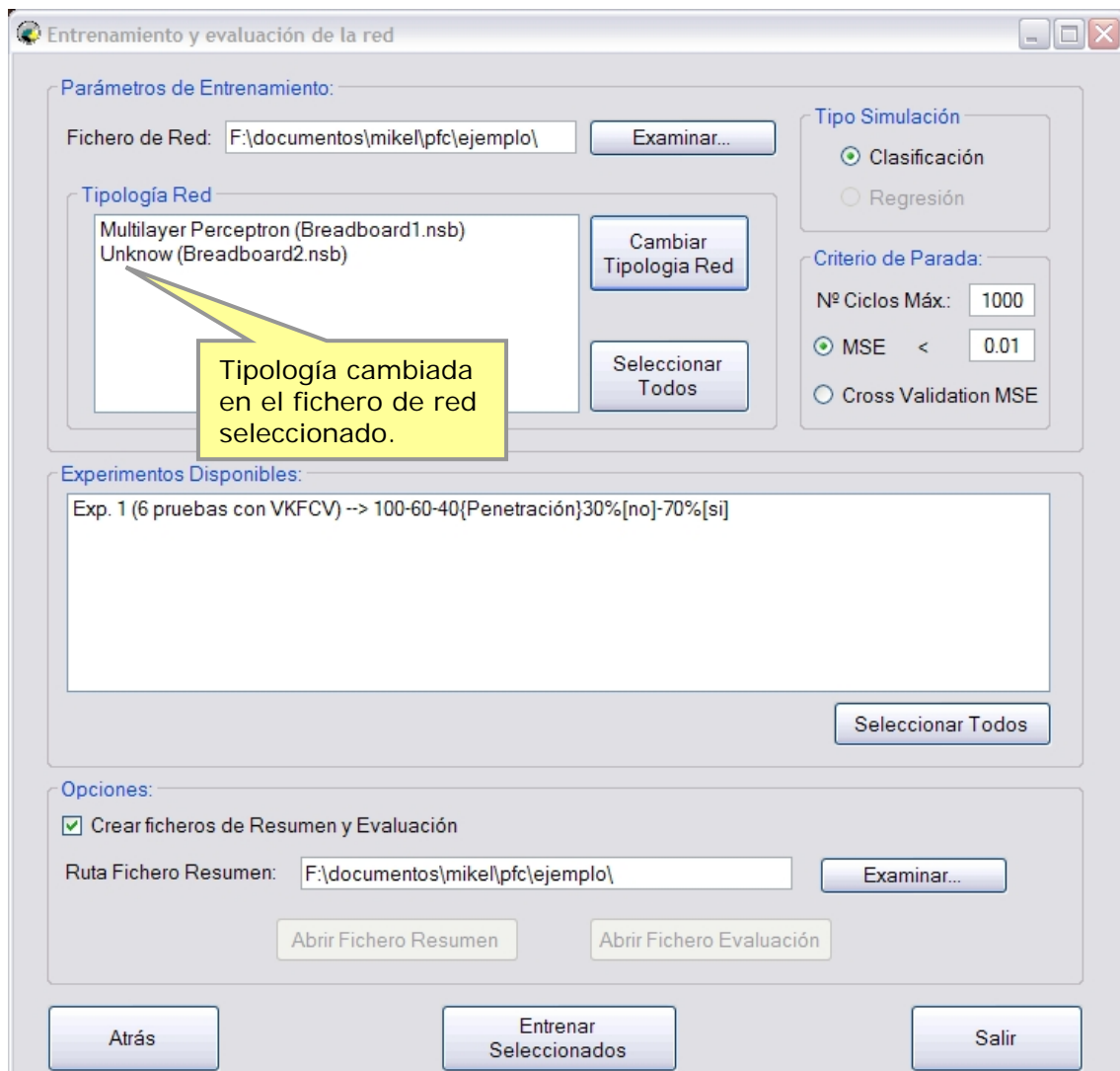


Figura 69. Resultado cambiar manualmente tipología de red



## **11.10 Configurar parámetros de entrenamiento**

El usuario puede modificar los parámetros que proporciona la aplicación por defecto ya sea el criterio de parada o la generación de informes. A continuación se define la configuración de los parámetros.

### **11.10.1 Criterio de parada**

La aplicación permite elegir entre varios criterios de parada para realizar el entrenamiento. Los criterios de parada son los siguientes:

- **Nº de ciclos máximos:** Este parámetro establece el número máximo de ciclos como criterio de parada para entrenar un experimento en la red. Por defecto su valor es 1000 y viene seleccionado. para poder modificarlo solo hay que escribir el valor deseado en el cuadro situado a la derecha del texto "Nº de ciclos máximos"
- **MSE:** Este parámetro indica como criterio de parada que el entrenamiento no supere el valor especificado de error cuadrático medio en un entrenamiento. Por defecto tiene un valor  $<0.01$ , para poder modificarlo solo hay que seleccionarlo y escribir el valor deseado en el cuadro situado a la derecha del texto "MSE"
- **Cross Validation MSE:** Este parámetro determina como criterio de parada el error cuadrático medio en base a la validación cruzada. Solo está habilitado cuando los experimentos tengan ficheros de validación cruzada.



En la siguiente figura podemos observar los criterios de parada en la aplicación:

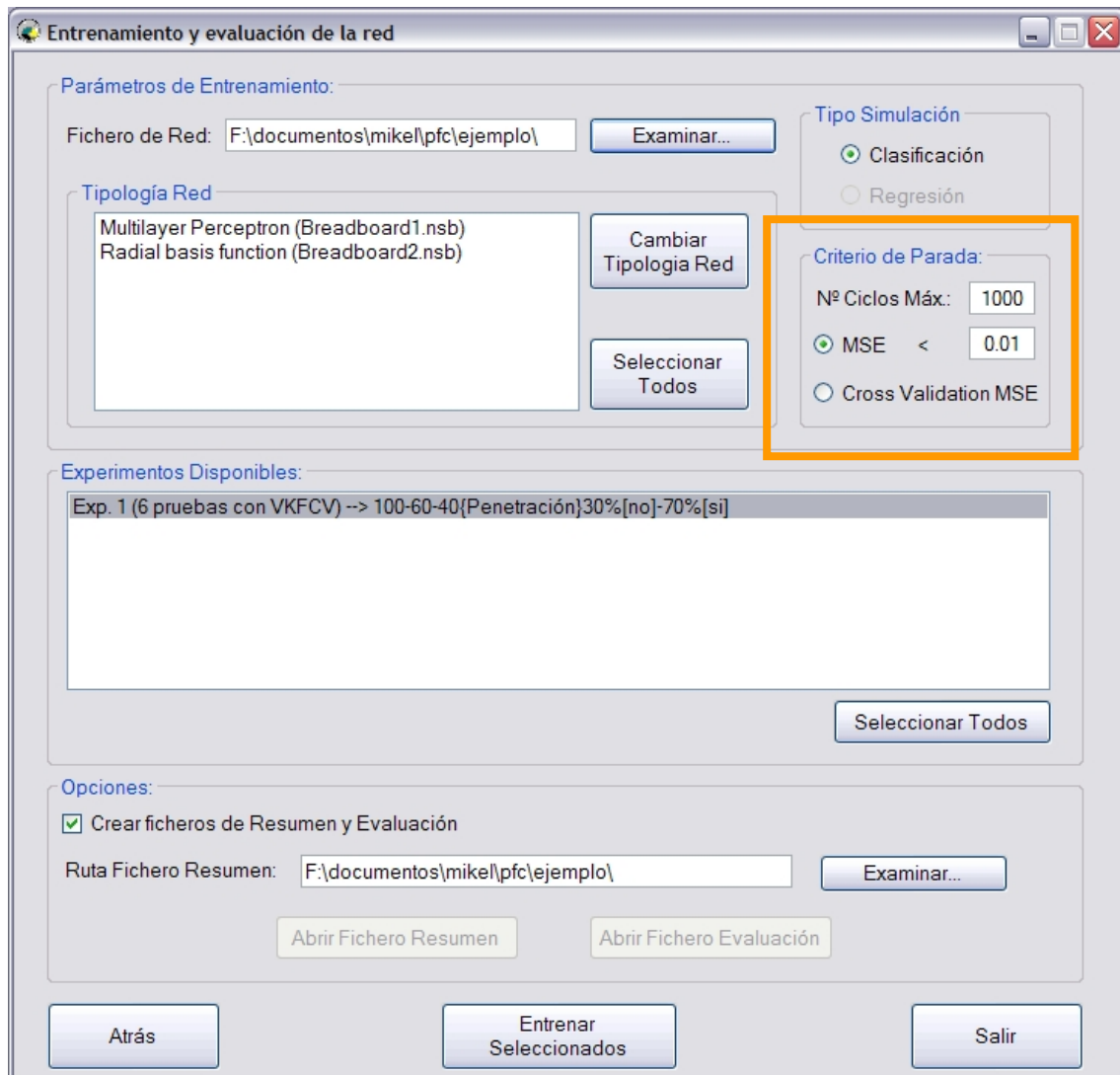
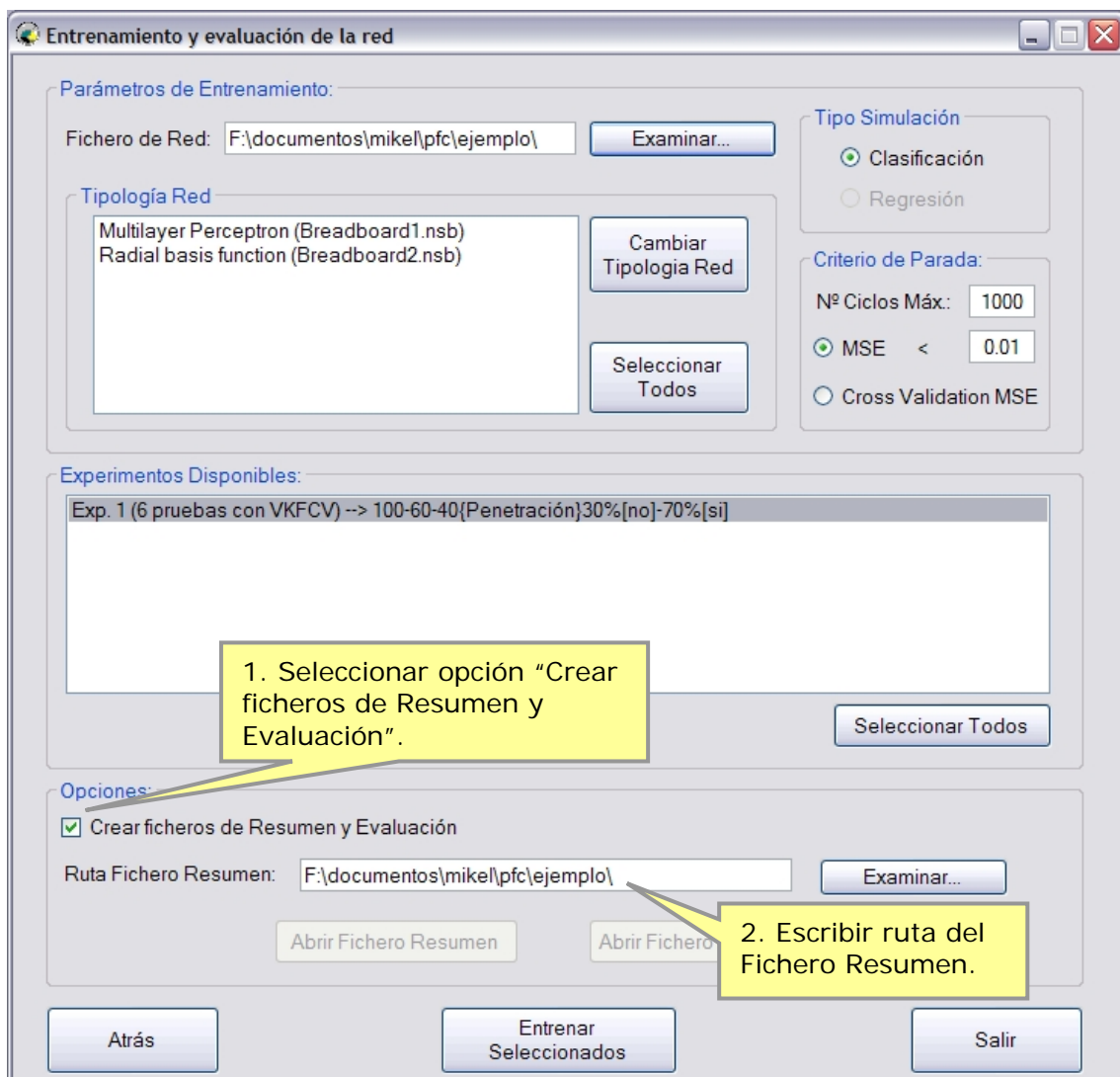


Figura 70. Criterios de parada

## 11.10.2 Generación de informes

La generación de Informes sobre el entrenamiento permite mostrar los resultados del entrenamiento de una forma más sencilla y completa.

Para que la aplicación genere los informes en el entrenamiento, el usuario deberá activar esta opción indicando la ruta del fichero resumen. Por defecto la aplicación establece la ruta del fichero resumen la misma que esta establecida para los ficheros de datos. El usuario puede cambiar la ruta simplemente sobrescribiendo en el cuadro de texto a la derecha del texto "Ruta Fichero Resumen".



**Entrenamiento y evaluación de la red**

**Parámetros de Entrenamiento:**

Fichero de Red: F:\documentos\mikel\pfc\ejemplo\ Examinar...

**Tipología Red**

Multilayer Perceptron (Breadboard1.nsb)  
Radial basis function (Breadboard2.nsb)

Cambiar Tipología Red

Seleccionar Todos

**Tipo Simulación**

☒ Clasificación  
☐ Regresión

**Criterio de Parada:**

Nº Ciclos Máx.: 1000

☒ MSE < 0.01  
☐ Cross Validation MSE

**Experimentos Disponibles:**

Exp. 1 (6 pruebas con VKFCV) --> 100-60-40{Penetración}30%{no}-70%{si}

Seleccionar Todos

**Opciones:**

☒ Crear ficheros de Resumen y Evaluación

Ruta Fichero Resumen: F:\documentos\mikel\pfc\ejemplo\ Examinar...

Abrir Fichero Resumen Abrir Fichero

Atrás Entrenar Seleccionados Salir

1. Seleccionar opción "Crear ficheros de Resumen y Evaluación".

2. Escribir ruta del Fichero Resumen.

Figura 71. Fichero evaluación y resumen

El usuario también puede introducir la ruta de la creación del fichero resumen haciendo clic sobre el botón “Examinar...” que se encuentra a la derecha del texto “Ruta Fichero Resumen”. La aplicación mostrará un cuadro de dialogo para establecer la ruta. En la siguiente figura se puede apreciar el proceso mencionado:

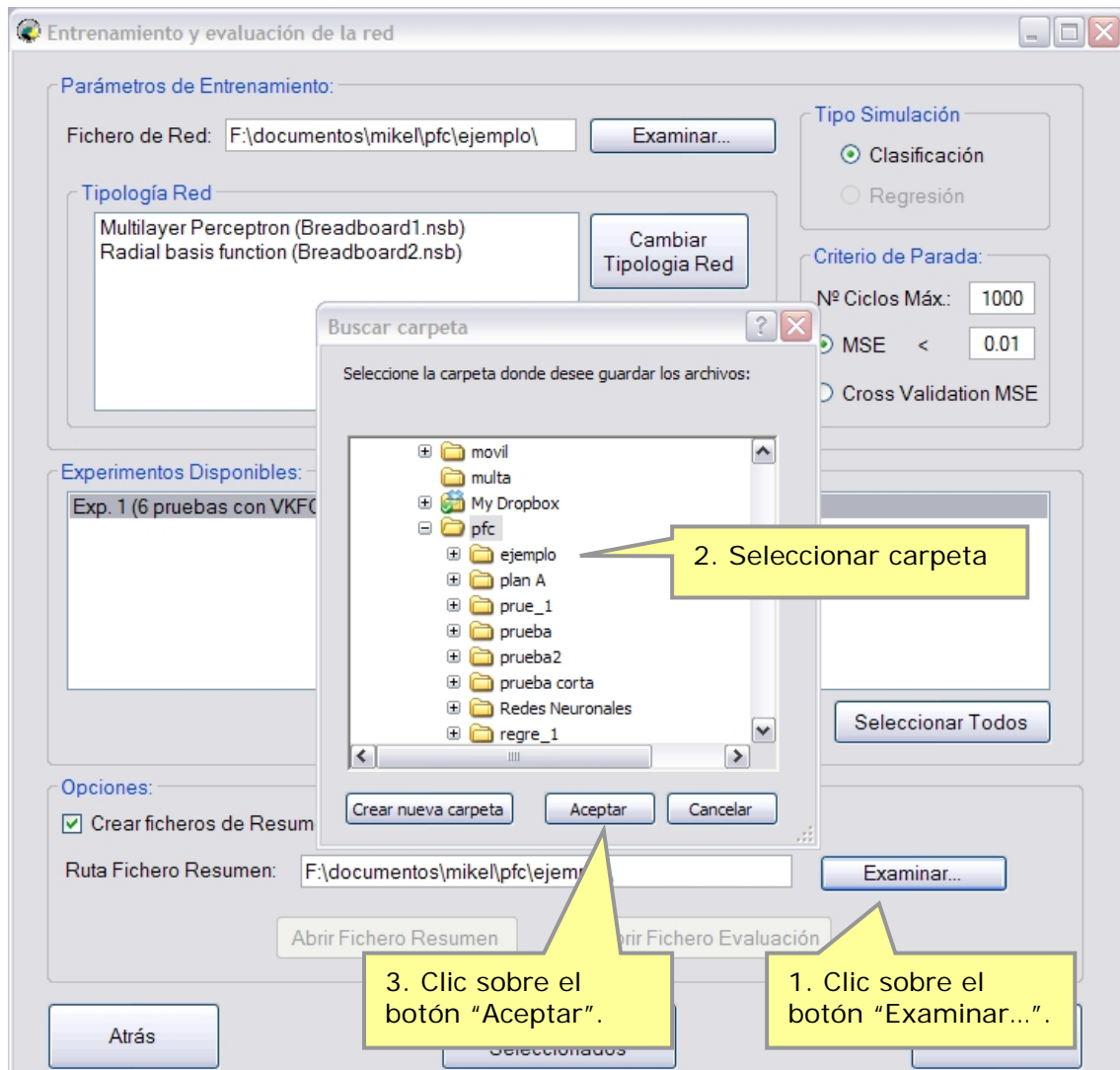


Figura 72. Establecer ruta fichero resumen

Nota: Para la realización de los informes la aplicación tiene en cuenta si los experimentos son de evaluación o regresión.

## 11.11 Entrenamiento

Una vez configurados todos los parámetros y opciones del entrenamiento, procederemos realizar el entrenamiento.

Primero debemos seleccionar la red o redes donde queremos realizar el entrenamiento de la lista "tipología Red".

Luego seleccionamos los experimentos que deseemos de la lista "Experimentos Disponibles"

NOTA: Para seleccionar varios elementos usar CONTROL+CLIC, o si seleccionamos todos los elementos podemos pulsar el botón "seleccionar todos" a la derecha de la lista.

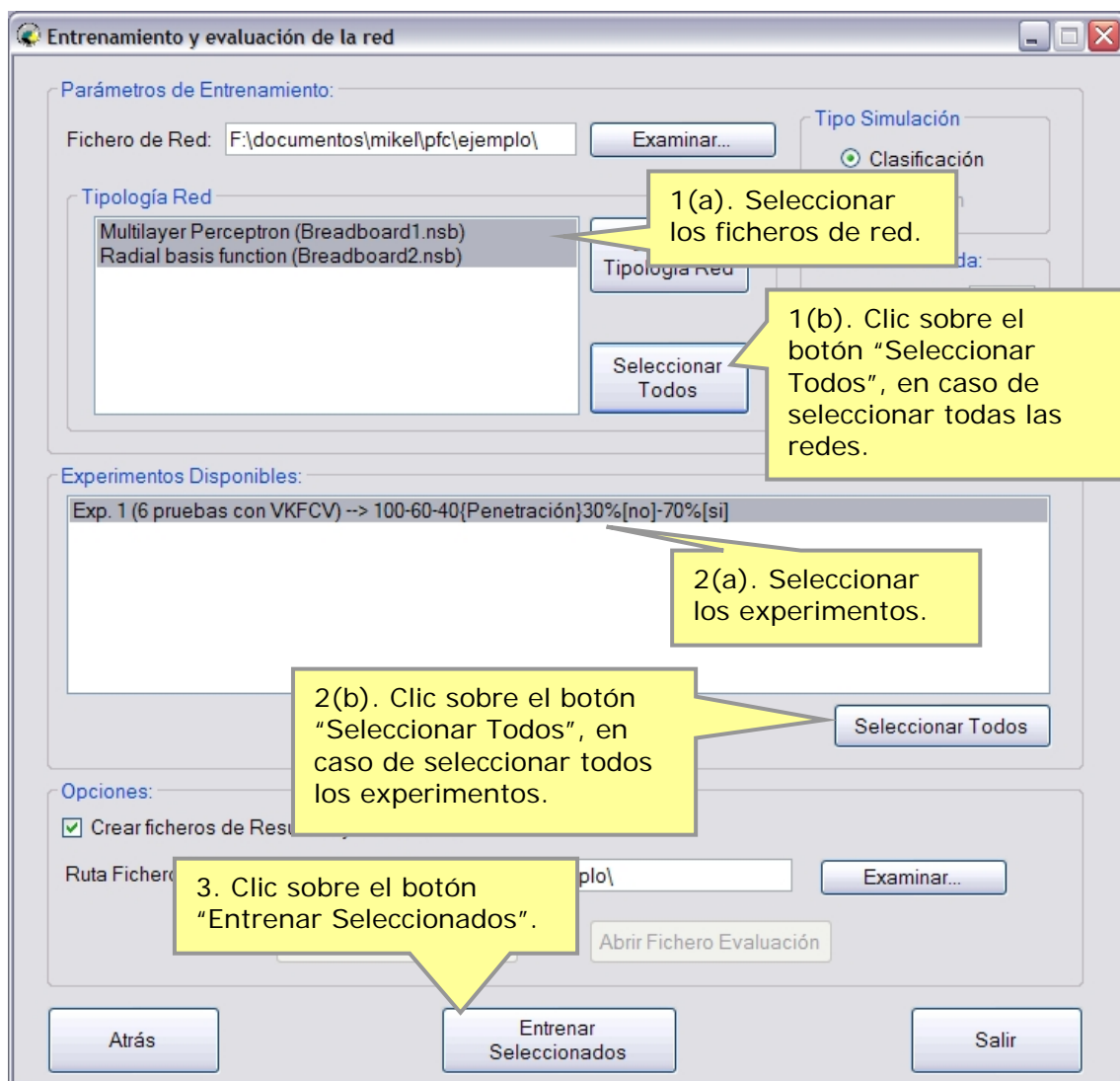


Figura 73. Seleccionar elementos entrenamiento

Una vez seleccionado todos los elementos hacemos clic sobre el botón “Entrenar Seleccionados” para comenzar el entrenamiento.

A continuación la aplicación realizará el experimento junto a la aplicación Neurosolutions. En la siguiente figura vemos como la aplicación muestra un cuadro de dialogo que informa del estado del entrenamiento.

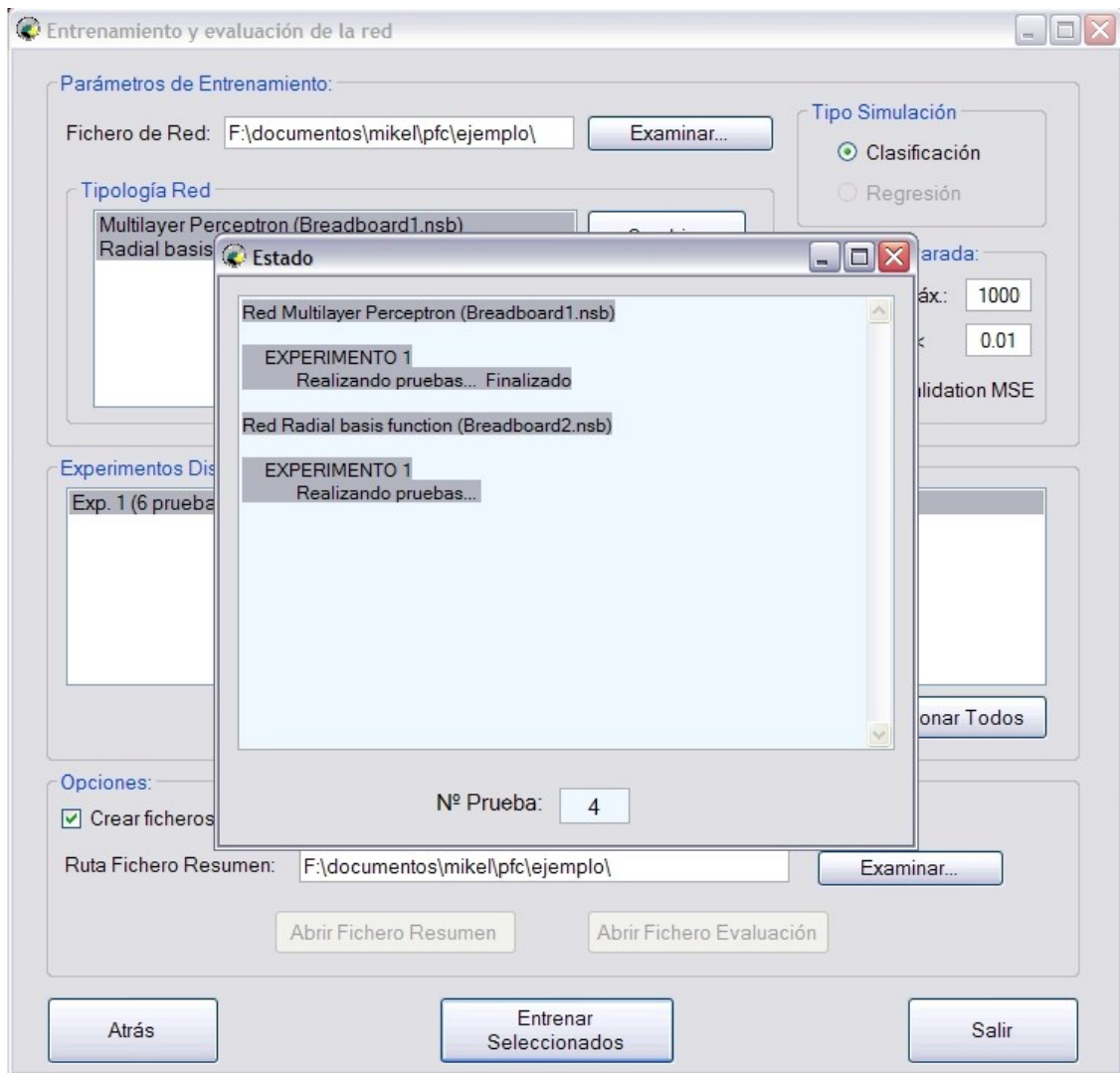


Figura 74.Estado entrenamiento

Una vez finalizado el entrenamiento la aplicación cerrará la aplicación neurosolutions y se dispondrá a realiza la generación de los informes (En caso de que hayamos seleccionado esta opción). En la siguiente figura podemos ver el cuadro de dialogo que se muestra mientras se generan los ficheros:

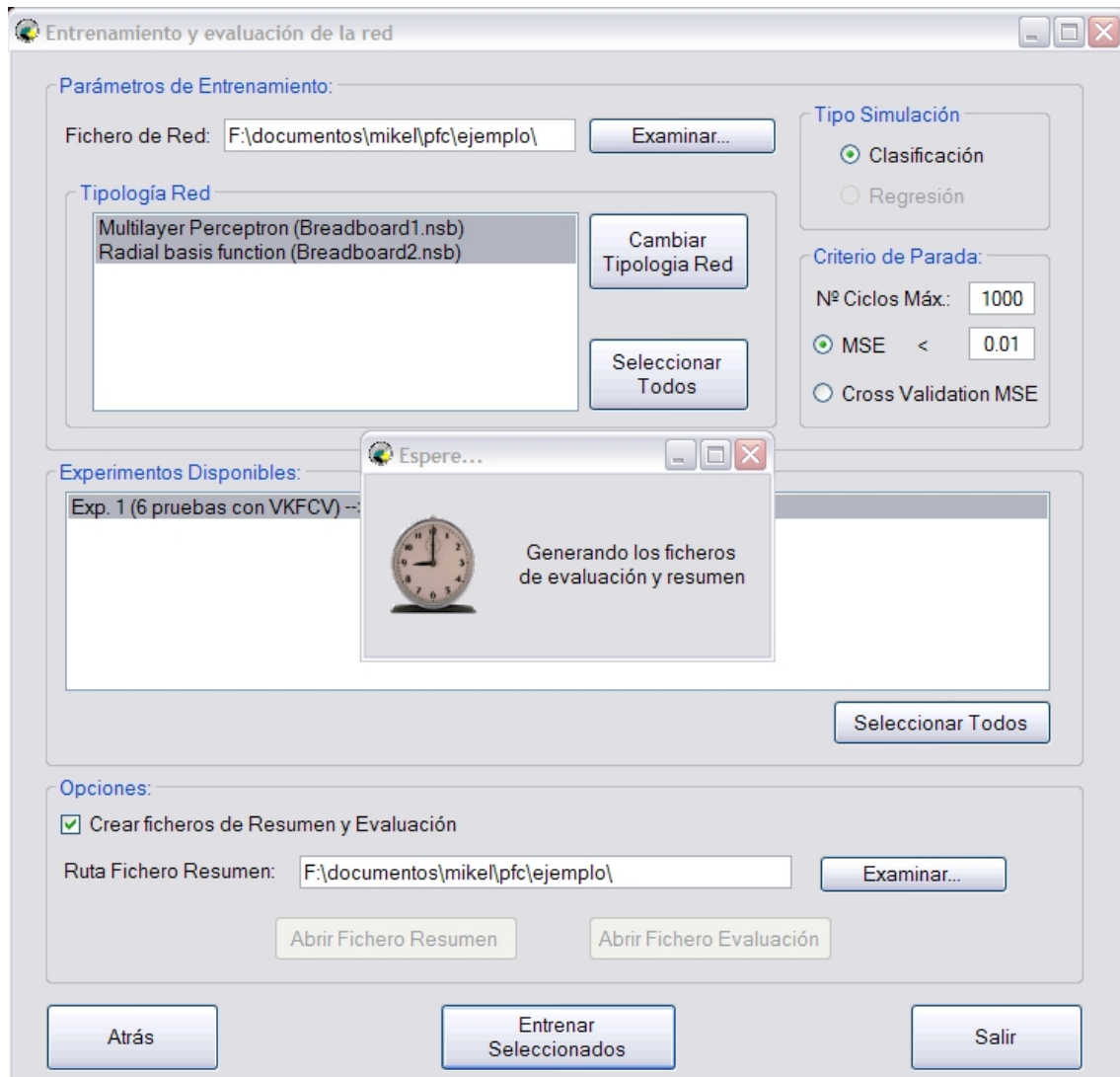


Figura 75. Generación de ficheros de evaluación y resumen

## 11.12 Informes de evaluación y resumen

Una vez terminado el entrenamiento, si anteriormente hemos habilitado la opción "Crear Ficheros de Resumen y Evaluación", podremos abrir desde la aplicación los correspondientes informes generados.

### 11.12.1 Informes de Evaluación

Se generarán tantos informes de evaluación como experimentos hayamos entrenado. Para abrir un informe seleccionaremos un experimento de la lista "Experimentos disponibles" y haremos clic en el botón "Abrir fichero Evaluación"

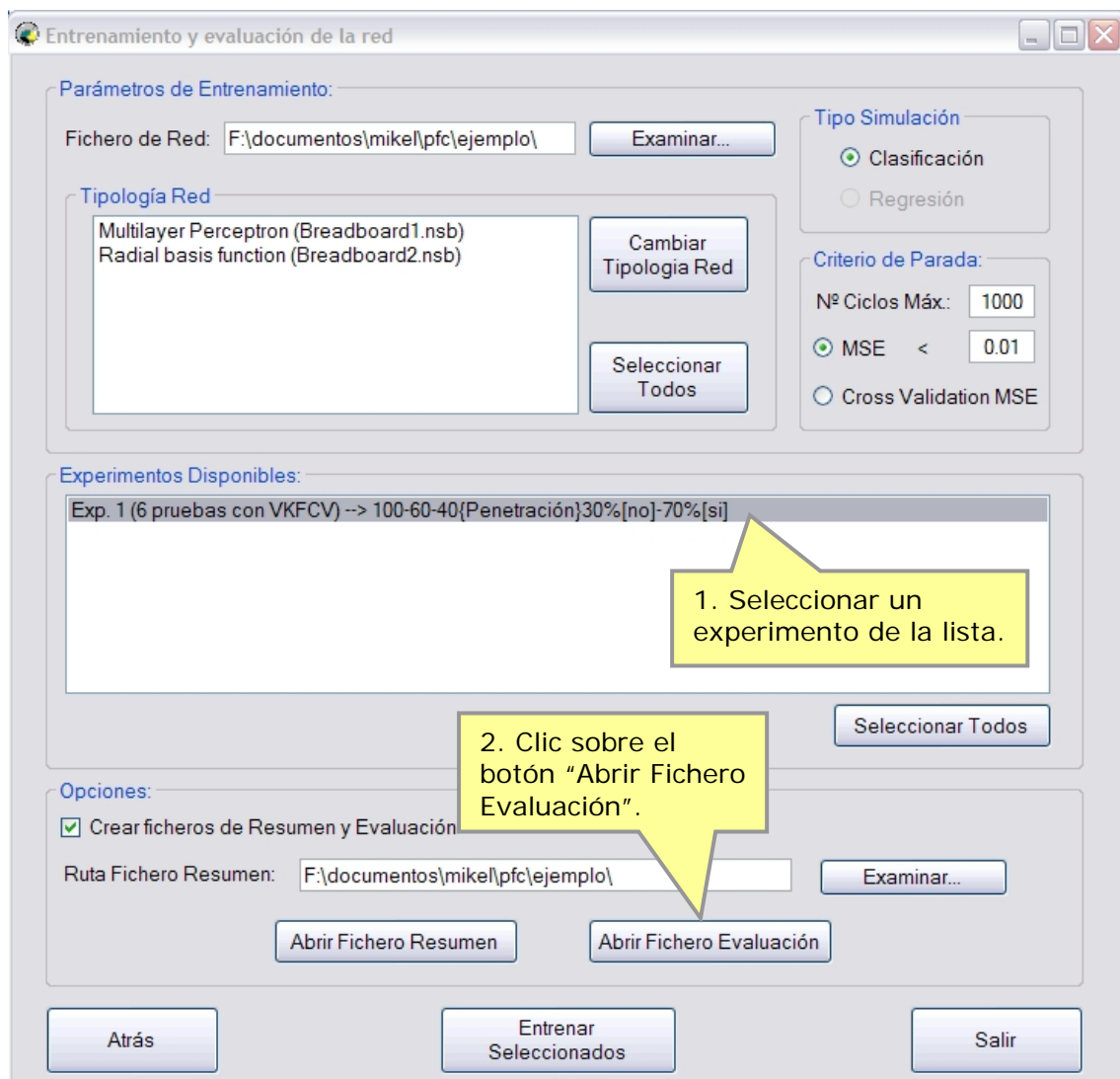
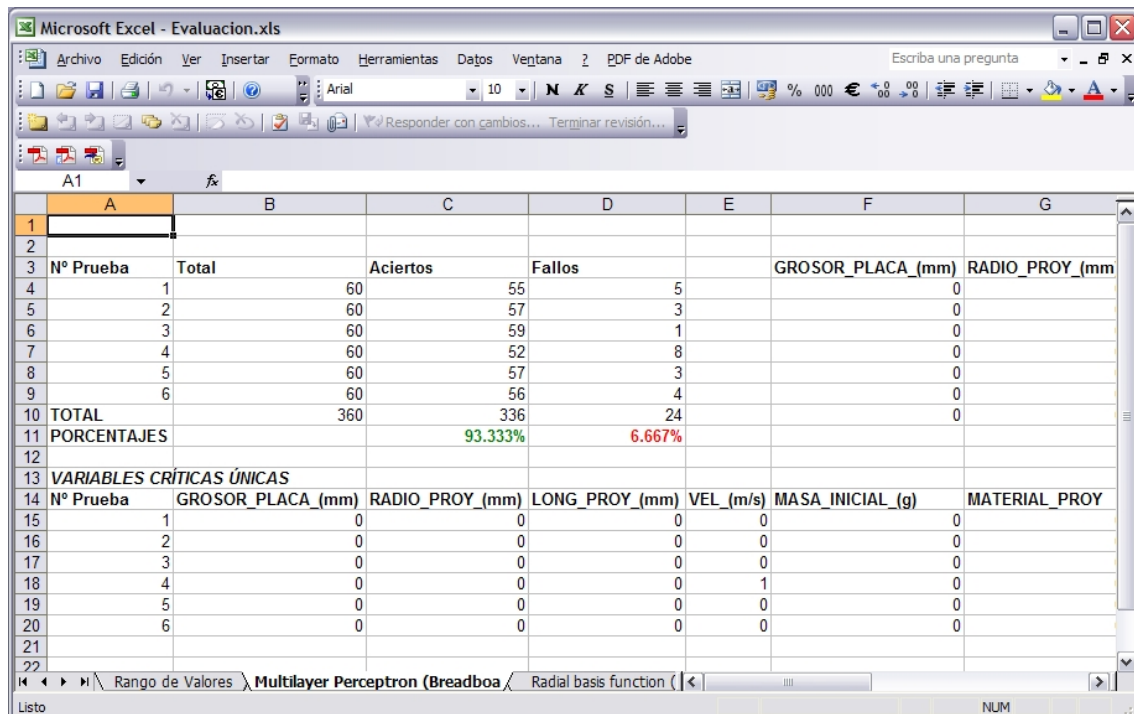


Figura 76. Abrir Informe de evaluación



El Informe de Evaluación se genera en formato de libro de Excel. Contiene una hoja llamada "Rango de Valores" que contiene el valor mínimo y máximo de cada parámetro de entrada en cada prueba. Además contiene tantas hojas como redes hayan participado en el entrenamiento. El nombre de cada hoja es el nombre lleva la estructura *tipología red (nombre fichero)*.

El informe de evaluación contiene información estadística extraída de las salidas del entrenamiento. Esta información es distinta si el entrenamiento es de clasificación o regresión. En las siguientes imágenes podemos apreciar las diferencias:



	A	B	C	D	E	F	G
1							
2							
3	Nº Prueba	Total	Aciertos	Fallos		GROSOR_PLACA_(mm)	RADIO_PROY_(mm)
4	1	60	55	5		0	
5	2	60	57	3		0	
6	3	60	59	1		0	
7	4	60	52	8		0	
8	5	60	57	3		0	
9	6	60	56	4		0	
10	TOTAL	360	336	24		0	
11	PORCENTAJES		93.333%	6.667%			
12							
13	VARIABLES CRÍTICAS ÚNICAS						
14	Nº Prueba	GROSOR_PLACA_(mm)	RADIO_PROY_(mm)	LONG_PROY_(mm)	VEL_(m/s)	MASA_INICIAL_(g)	MATERIAL_PROY
15	1	0	0	0	0	0	
16	2	0	0	0	0	0	
17	3	0	0	0	0	0	
18	4	0	0	0	1	0	
19	5	0	0	0	0	0	
20	6	0	0	0	0	0	
21							
22							

Figura 77. Informe evaluación entrenamiento clasificación





## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Microsoft Excel - Evaluacion.xls

Archivo Edición Ver Insertar Formato Herramientas Datos Ventana 2 PDF de Adobe

Escriba una pregunta

Responder con cambios... Terminar revisión...

	A1	Nº Prueba										
	A	B	C	D	E	F	G	H	I	J	K	
1	Nº Prueba	Nº Tests	Masa_Final	Vres (m/s)								
2	1	40	0.94915666	0.76665059								
3	2	40	0.94359569	0.75989001								
4	TOTAL	80	0.94637618	0.7632703								
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												
17												
18												
19												
20												
21												
22												
23												
24												
25												
26												
27												
28												
29												
30												

Radial basis function (Breadboa / Multilayer Perceptron (Breadboa / Jordam-Eln

Listo NUM

Figura 78. Informe Evaluación entrenamiento regresión

## 11.12.2 Informe de Resumen

En un entrenamiento se genera un único informe de resumen. Para abrir el fichero haremos clic sobre el botón “Abrir Fichero Resumen”. Se abrirá el fichero que esta en formato de libro de Excel.

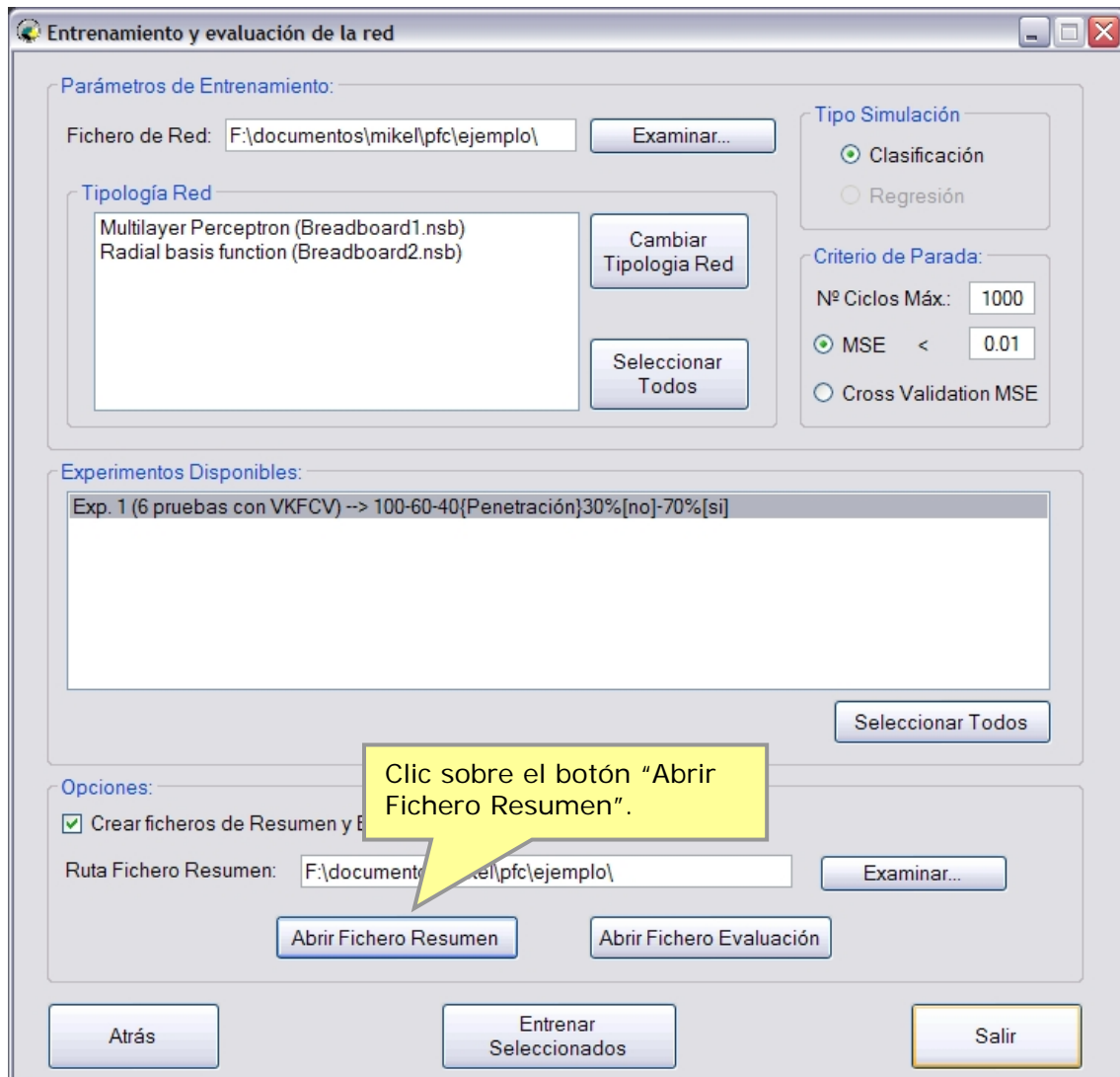


Figura 79. Abrir fichero informe de resumen.

El fichero contiene un resumen de información estadística de todos los experimentos que se han entrenado.

Al igual que los informes de evaluación la información mostrada depende del tipo de entrenamiento: clasificación o regresión. A continuación mostramos dos ficheros de resumen distintos:



## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Microsoft Excel - Resumen.xls

Archivo Edición Ver Insertar Formato Herramientas Datos Ventana ? PDF de Adobe

Escriba una pregunta

A1 TIPOLOGÍA

	A	B	C	D	E	F	G	H	I
1	TIPOLOGÍA	PRUEBAS	TRAIN	TEST	ACIERTOS	FALLOS	TOTAL	%ACIERTOS	%FALLOS
2	Multilayer Pe	6	100	60	336	24	360	93.333%	6.667%
3	Radial basis	6	100	60	318	42	360	88.333%	11.667%
4									
5									
6	TIPOLOGÍA	PRUEBAS	TRAIN	TEST	GROSOR_PLACA_(mm)	RADIO_PROY_(mm)	LONG_PROY_(mm)	VEL_(m/s)	MASA_INICIAL_(g)
7	Multilayer Pe	6	100	60	0	0	0	0.333333333	0
8	Radial basis	6	100	60	0	0	0	0	0
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									

Resumen / Hoja2 / Hoja3 /

Listo NUM

Figura 80. Informe resumen entrenamiento clasificación

Microsoft Excel - Resumen.xls

Archivo Edición Ver Insertar Formato Herramientas Datos Ventana ? PDF de Adobe

Escriba una pregunta

A1 TIPOLOGÍA RED

	A	B	C	D	E	F	G	H	I	J	K
1	TIPOLOGÍA RED	PRUEBAS	TRAIN	TEST	TOTAL	Media Masa_Final_(g)	Media Vres_(m/s)				
2	Radial basis functi	5	50	40	200	0.85899173	0.815494563				
3	Multilayer Percept	5	50	40	200	0.670078537	0.604817872				
4	Jordam-Elman (jor	5	50	40	200	0.919167933	0.845080443				
5	Multilayer Percept	5	50	40	200	0.945487393	0.878378768				
6	Support Vector Ma	5	50	40	200	0.576095049	0.681916272				
7	Radial basis functi	4	50	40	160	0.857474596	0.85737102				
8	Multilayer Percept	4	50	40	160	0.573217241	0.299202912				
9	Jordam-Elman (jor	4	50	40	160	0.92321366	0.813142295				
10	Multilayer Percept	4	50	40	160	0.944328522	0.917885521				
11	Support Vector Ma	4	50	40	160	0.700780591	0.689837235				
12	Radial basis functi	3	50	40	120	0.93249768	0.876257195				
13	Multilayer Percept	3	50	40	120	0.612462352	0.467910209				
14	Jordam-Elman (jor	3	50	40	120	0.939666854	0.884899365				
15	Multilayer Percept	3	50	40	120	0.92376609	0.904969962				
16	Support Vector Ma	3	50	40	120	0.779098271	0.75945106				
17	Radial basis functi	2	50	40	80	0.946376177	0.763270303				
18	Multilayer Percept	2	50	40	80	0.556503571	0.443531063				
19	Jordam-Elman (jor	2	50	40	80	0.951034647	0.877052626				
20	Multilayer Percept	2	50	40	80	0.956611975	0.894878069				
21	Support Vector Ma	2	50	40	80	0.580991682	0.728843596				
22											
23											
24											
25											
26											
27											
28											
29											
30											

Resumen / Hoja2 / Hoja3 /

Listo NUM

Figura 81. Informe resumen entrenamiento regresión



### 11.13 Salir de la aplicación

Para poder abandonar la aplicación se debe hacer clic en el botón "salir".

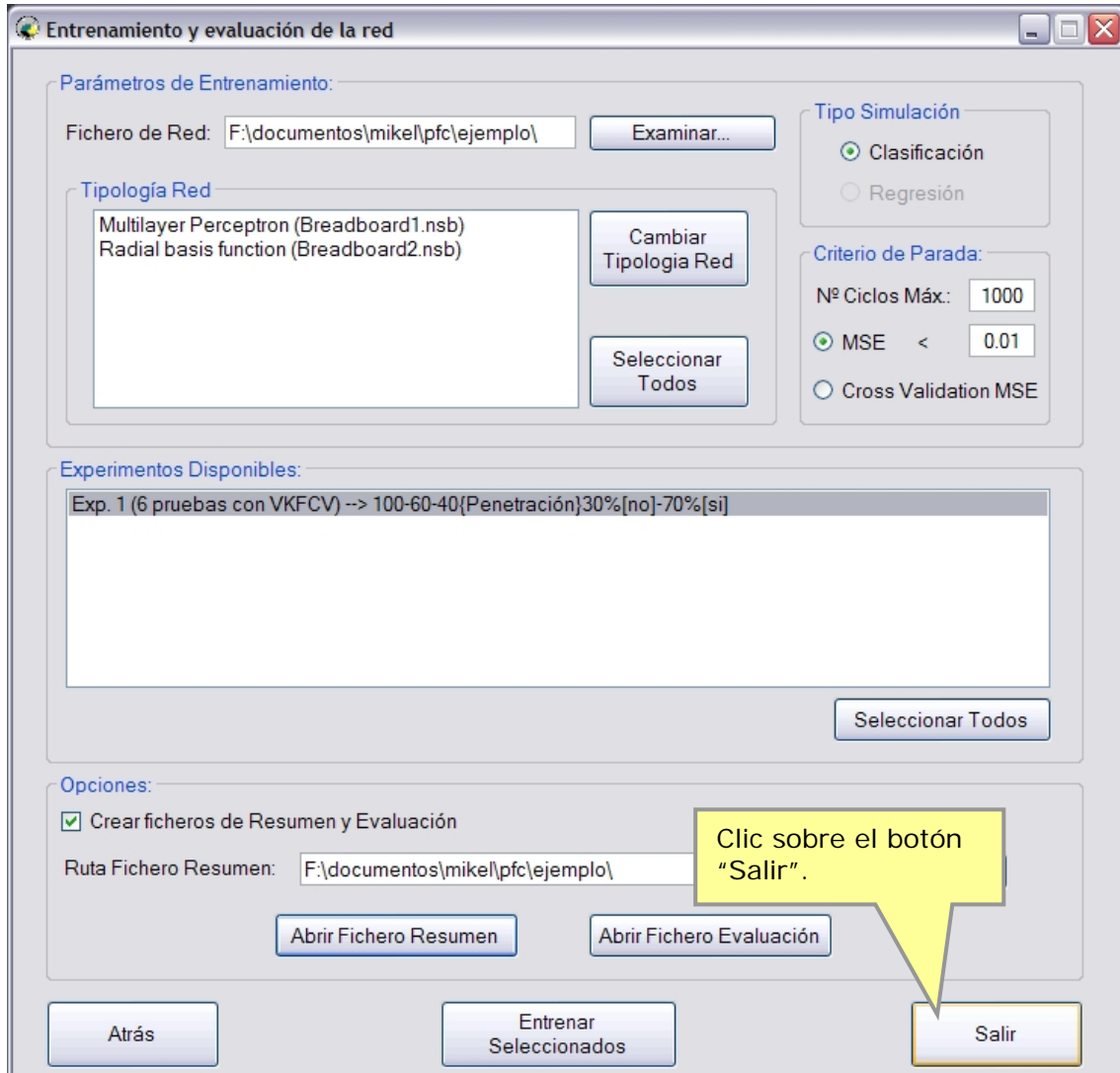


Figura 82. Salir de la aplicación



## **12 Anexo B: Formato para ficheros de entrada**

La aplicación requiere para su correcto funcionamiento que los ficheros de entrada empleados cumplan los requisitos enumerados a continuación:

El fichero que incluya los datos a procesar debe ser un fichero creado por Microsoft Excel con la extensión “.xls”.

La fila número uno de la hoja donde se encuentren los datos debe contener los nombres de los atributos en cada una de las columnas.

Cada atributo se encontrará en una columna de la hoja, empezando por la columna uno de la hoja, añadiendo una columna para cada atributo de forma consecutiva a la anterior.

Cada instancia estará representada en una fila de la hoja, empezando por la fila dos de la hoja, añadiendo una fila por cada instancia de forma consecutiva a la anterior.

Es recomendable que el fichero no contenga macros o similares, ya que podría darse el caso de ser incompatibles con el correcto funcionamiento de la aplicación.

## 13 Anexo C. Manual de usuario para la creación de una red neuronal en NeuroSolutions

Para crear una nueva red, se inicia la aplicación NeuroSolutions y se hace clic sobre el botón "NBuilder" indicado en la figura:

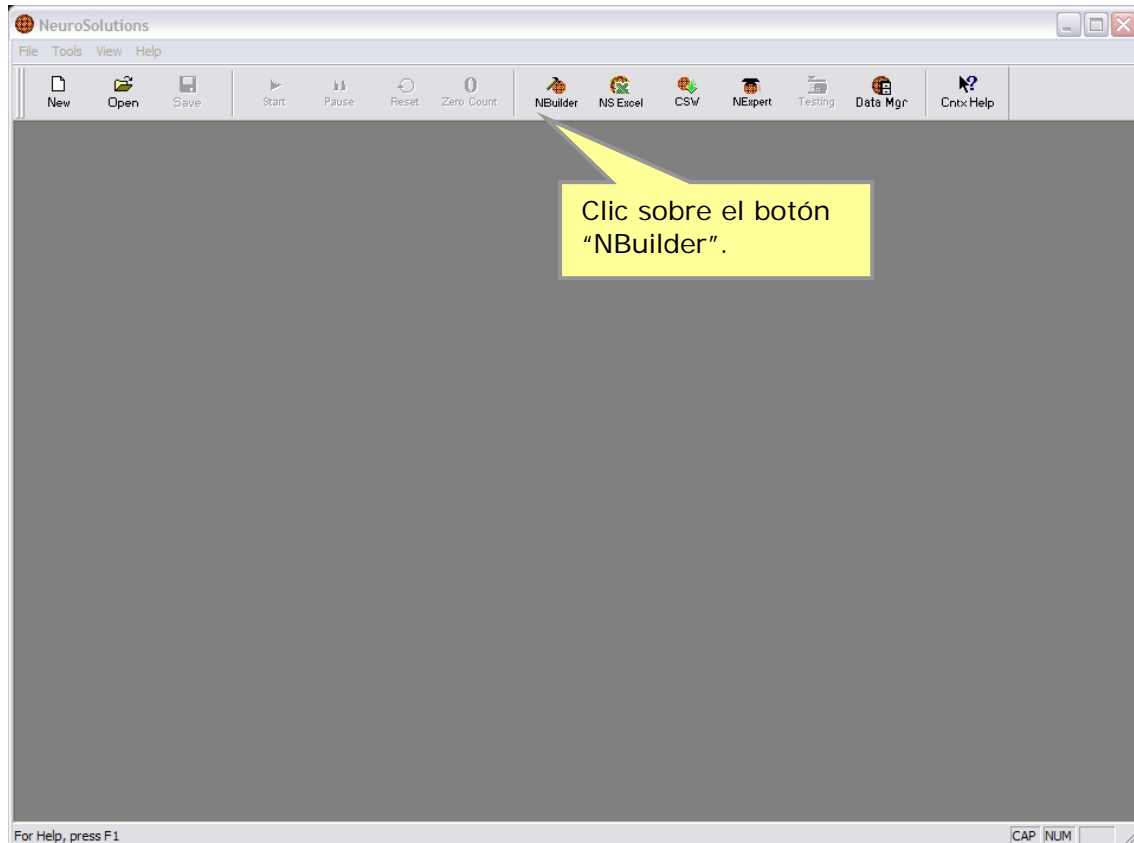


Figura 83. Aplicación Neurosolutions

A continuación se iniciará la aplicación "neuralbuilder". Que nos permitirá crear una red con una tipología específica.

En este manual generaremos una red perceptron multicapas, el proceso es similar para el resto de tipológicas.

En la siguiente figura podemos observar la ventana donde elegiremos el tipo de red a generar:

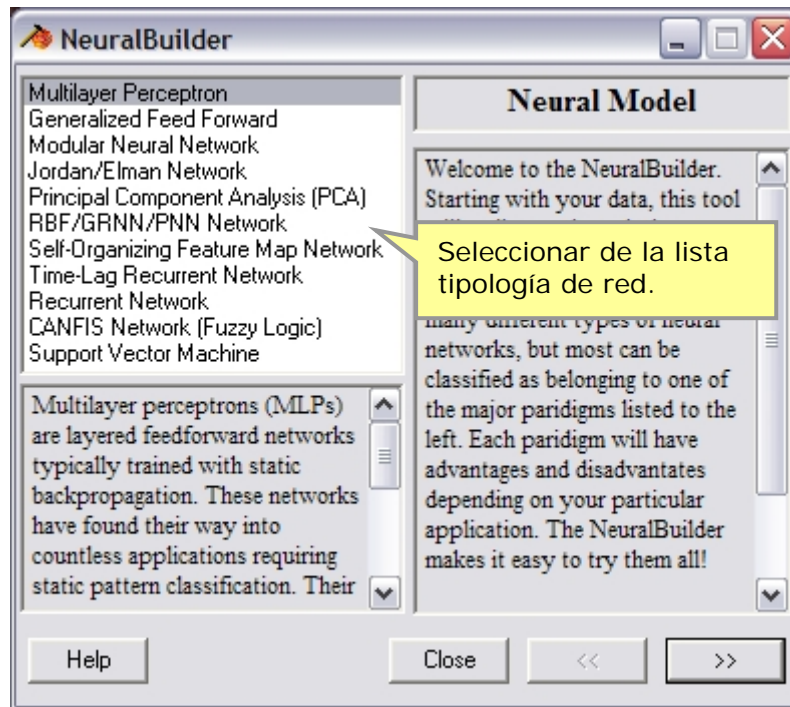


Figura 84. Elección tipología red en Neural builder

Seleccionamos la tipología que queremos generar y hacemos clic sobre el botón ">>" para continuar

En la siguiente ventana debemos introducir los datos de entrenamiento (Train), para ello hacemos clic sobre el botón "Browse..."

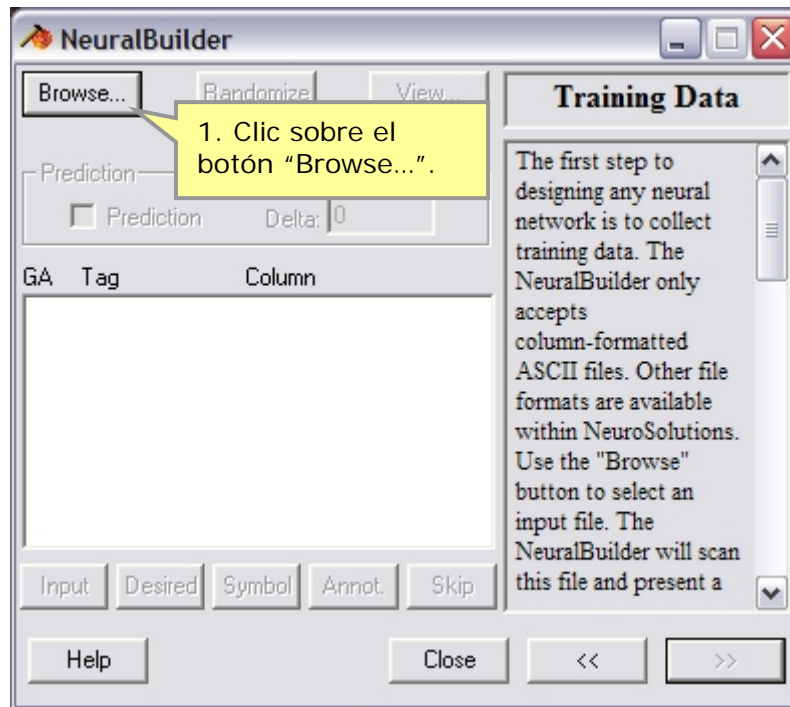


Figura 85. Introducción fichero de entrenamiento



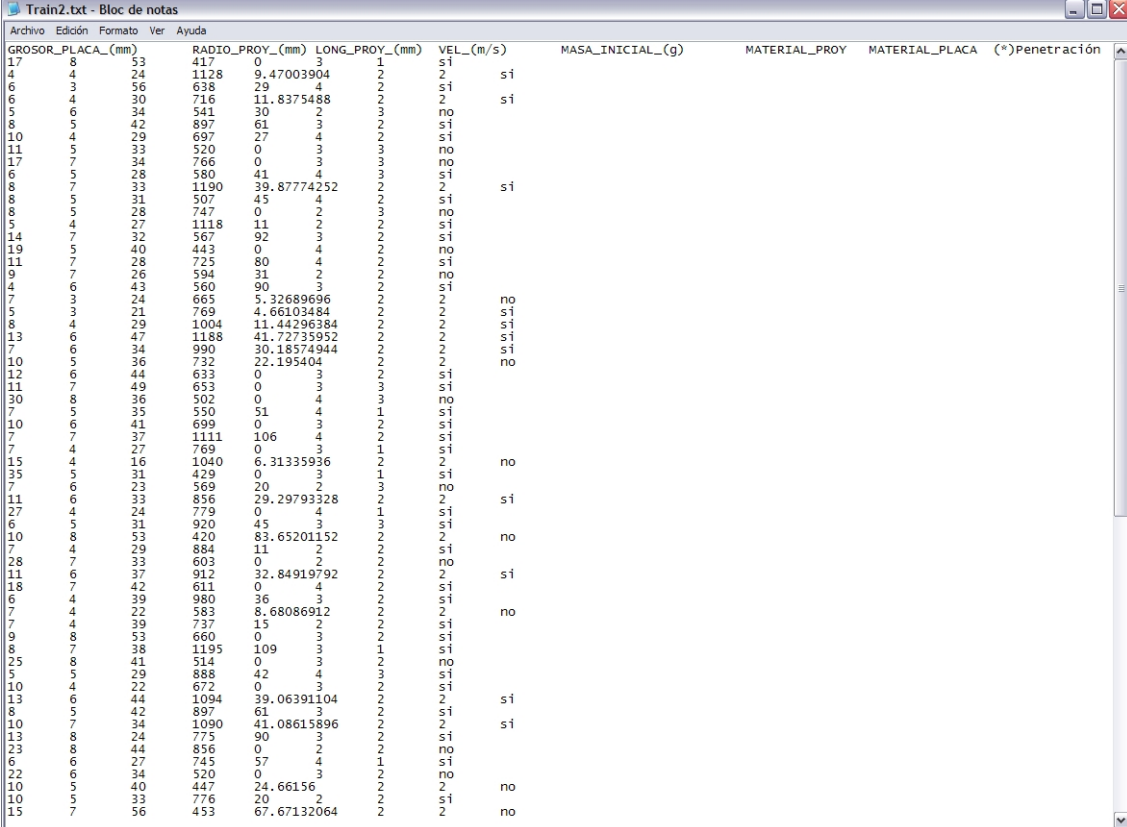


## Proyecto Fin de Carrera

Miguel Fortes Gómez

Ingeniería Técnica en Informática de gestión

Para crear la red desde el propio NeuroSolutions, los ficheros donde se encuentran los datos deben ser ficheros ASCII, organizados en filas y cada variable en una columna, es decir, en formato de "tabla". Un ejemplo de fichero de datos sería el que se observa en la siguiente figura:



GROSOR_PLACA_(mm)	RADIO_PROY_(mm)	LONG_PROY_(mm)	VEL_(m/s)	MASA_INICIAL_(g)	MATERIAL_PROY	MATERIAL_PLACA	(*)Penetración
17	8	53	417	0	3	1	si
4	4	24	1128	9.47003904	2	2	si
6	3	56	638	29	4	2	si
6	4	30	716	11.8375488	2	2	si
5	6	34	541	30	2	3	no
8	5	42	897	61	3	2	si
10	4	29	697	27	4	2	si
11	5	33	520	0	3	3	no
17	7	34	766	0	3	3	no
6	5	28	580	41	4	3	si
8	5	33	1190	39.87774252	2	2	si
8	5	31	507	45	4	2	si
8	5	28	747	0	2	3	no
5	4	27	1118	11	2	2	si
14	7	32	567	92	3	2	si
19	5	40	443	0	4	2	no
11	7	28	725	80	4	2	si
9	7	26	594	31	2	2	no
4	6	43	560	90	3	2	si
7	3	24	665	5.32689696	2	2	no
5	3	21	769	4.66103484	2	2	si
8	4	29	1004	11.44296384	2	2	si
13	6	47	1188	41.72735952	2	2	si
7	6	34	990	30.18574944	2	2	si
10	5	36	732	22.195404	2	2	no
12	6	44	633	0	3	2	si
11	7	49	653	0	3	3	si
30	8	36	502	0	4	3	no
7	5	35	550	51	4	1	si
10	6	41	699	0	3	2	si
7	7	37	1111	106	4	2	si
7	4	27	769	0	3	1	si
15	4	16	1040	6.31335936	2	2	no
35	5	31	429	0	3	1	si
7	6	23	569	20	2	3	no
11	6	33	856	29.29793328	2	2	si
27	4	24	779	0	4	1	si
6	5	31	920	45	3	3	si
10	8	53	420	83.65201152	2	2	no
7	4	29	884	11	2	2	si
28	7	33	603	0	2	2	no
11	6	37	912	32.84919792	2	2	si
18	7	42	611	0	4	2	si
6	4	39	980	36	3	2	si
7	4	22	583	8.68086912	2	2	no
9	4	39	737	15	2	2	si
8	8	53	660	0	3	2	si
8	7	38	1195	109	3	1	si
25	8	41	514	0	3	2	no
5	5	29	888	42	4	3	si
10	4	22	672	0	3	2	si
13	6	44	1094	39.06391104	2	2	si
8	5	42	897	61	3	2	si
10	7	34	1090	41.08615896	2	2	si
13	8	24	775	90	3	2	si
23	8	44	856	0	2	2	no
6	6	27	745	57	4	1	si
22	6	34	520	0	3	2	no
10	5	40	447	24.66156	2	2	no
10	5	33	776	20	2	2	si
15	7	56	453	67.67132064	2	2	no

Figura 86. Fichero train

A continuación se deben etiquetar las variables que aparecen en el fichero, como entrada (Input), salida (Desired) o ignorada (Skip), dependiendo de las variables que se quieran utilizar en la simulación. Si alguna variable no es numérica se debe marcar como símbolo (Symbol) El resultado puede ser como el de la figura:

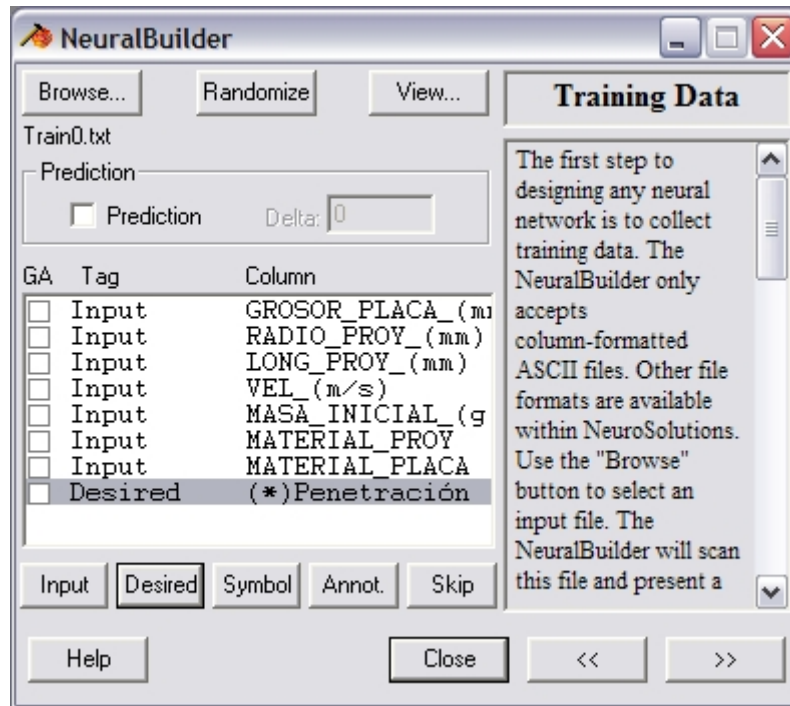


Figura 87. Parámetros fichero train

Se hace clic sobre el botón ">>", en la siguiente ventana debemos indicar el fichero de pruebas (test0.txt). Parar ello, basta con hacer clic en el botón "Input Test...".

En caso de que el experimento tenga validación cruzada debemos indicar el fichero de validación cruzada (val0.txt). Parar ello haremos clic en el botón "Input CV...".

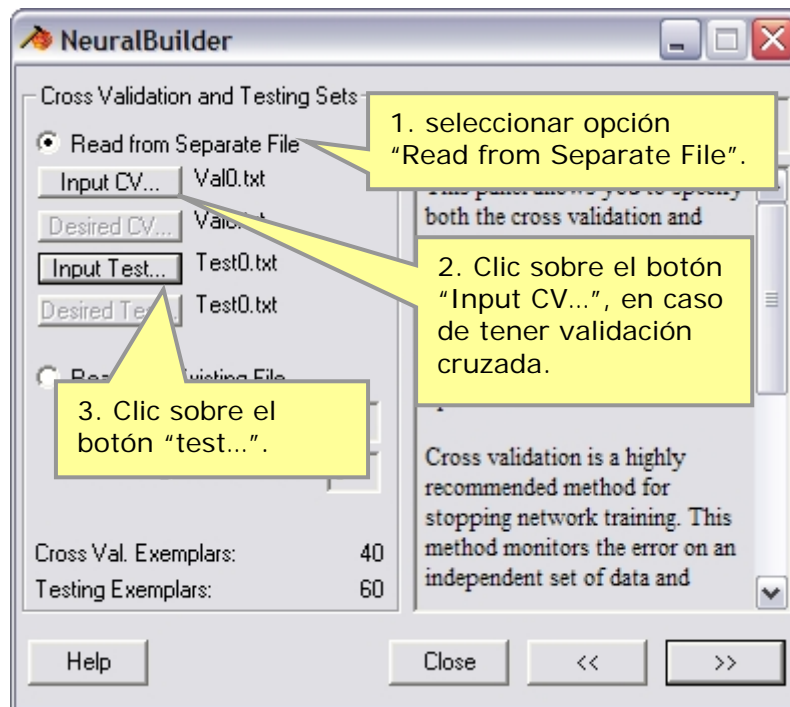


Figura 88. Selección fichero entrenamiento y validación cruzada

El siguiente paso es indicar el número de capas ocultas que se desea que tenga la red para la simulación, escribiéndolo en el apartado "Hidden Layers" como indica la figura:

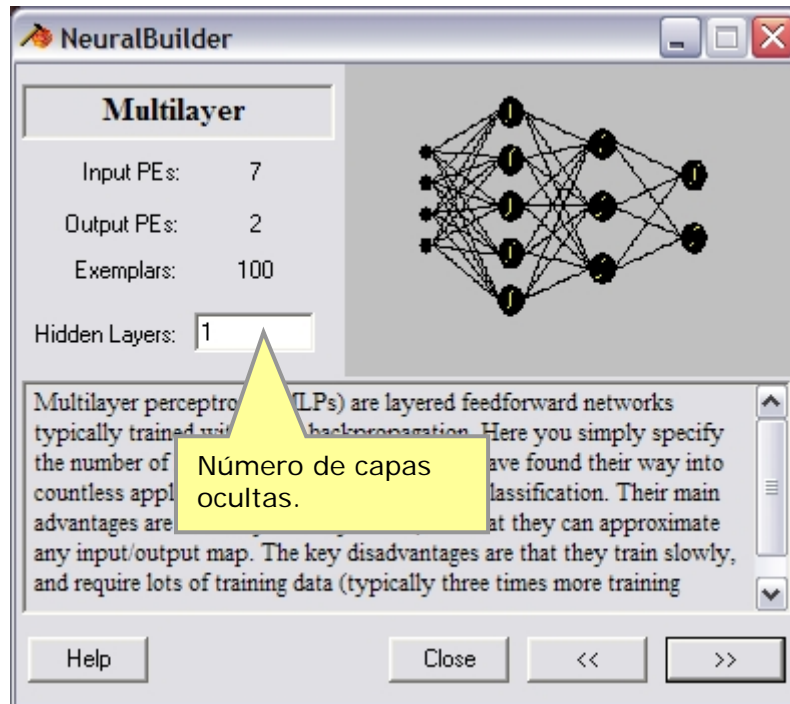


Figura 89. Número capas ocultas

Dependiendo del número de capas ocultas elegido a continuación saldrá una nueva ventana por cada una de ellas donde permite modificar algunos parámetros, aunque para este caso no es necesario. Un ejemplo para la capa oculta #1 sería:

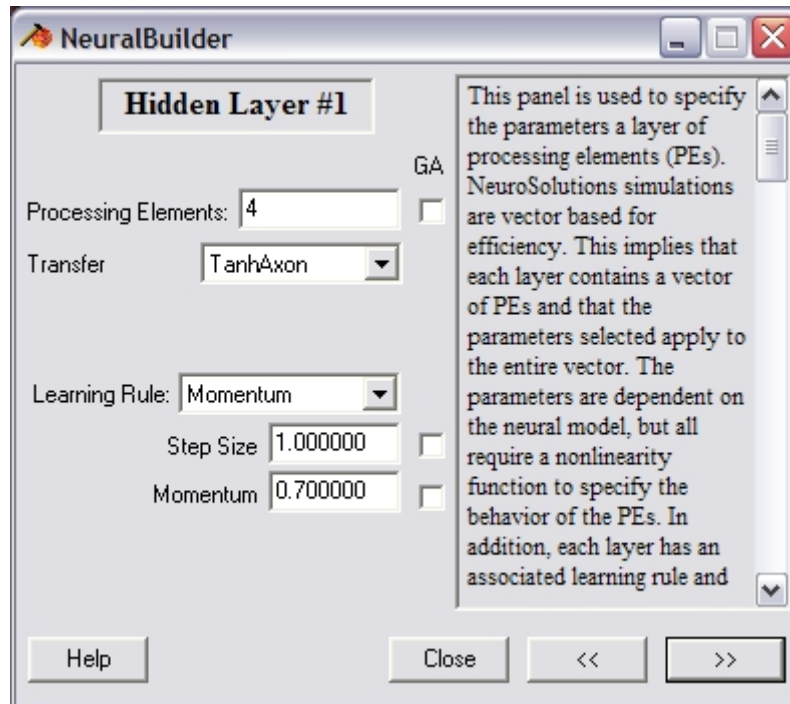


Figura 90. Número de neuronas ocultas

Una vez hecho clic en el botón ">>" de la última capa oculta, aparece una ventana para configurar la capa de salida de la red (en principio no es necesario modificar la configuración por defecto):

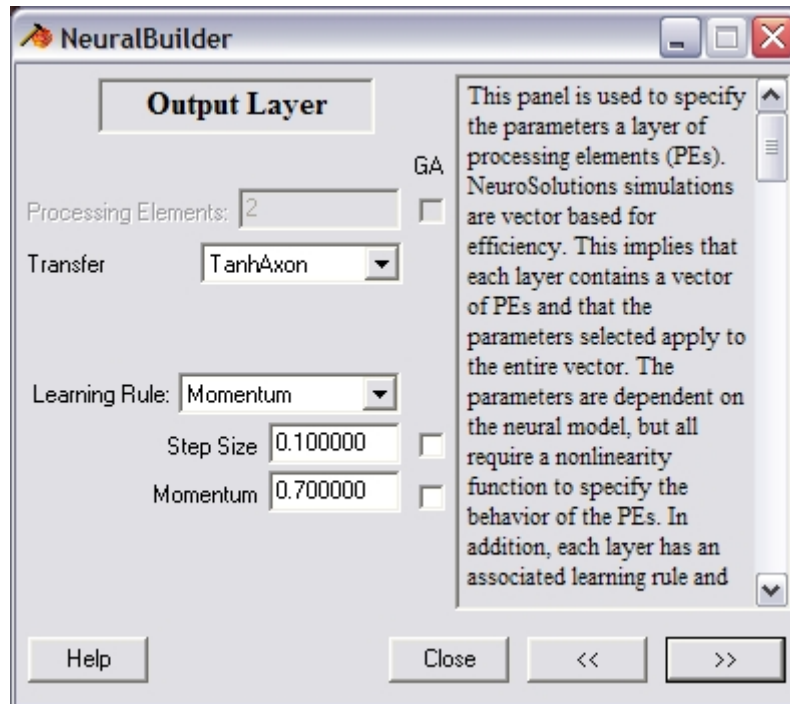


Figura 91. Configuración Capa de salida

El siguiente paso consiste en modificar el control de aprendizaje y el criterio de parada de la red. Sin embargo, no es necesario modificar ninguna de estas opciones aquí, ya que la propia aplicación Asistente se encarga de ello como se detalla más adelante.

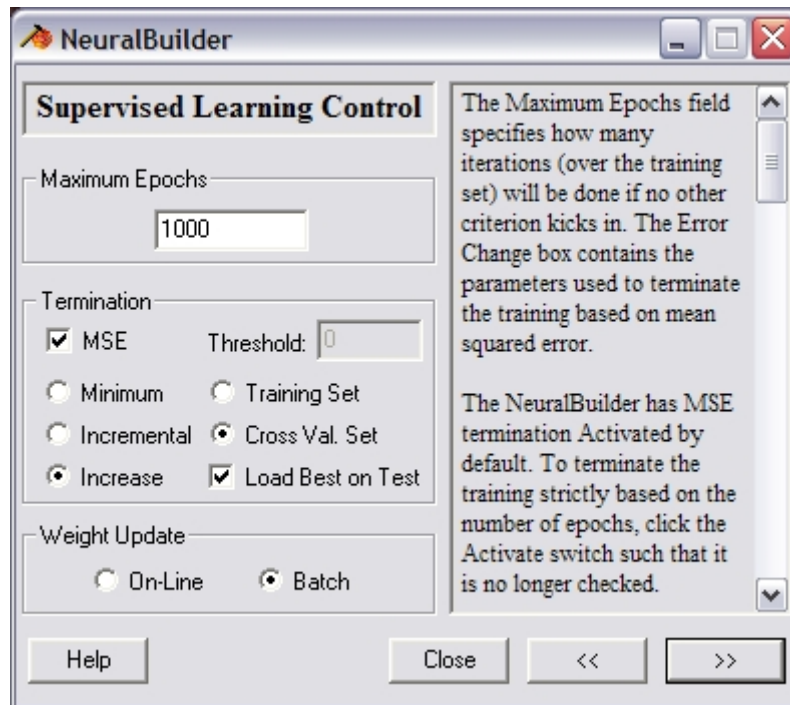


Figura 92 Configuración control de aprendizaje supervisado

Dependiendo de la tipología de la red podemos tener un control de aprendizaje supervisado, no supervisado o ambos.

Por último, la siguiente ventana permite configurar la salida de los resultados, y hacer clic en el botón “Build” para crear la red configurada:

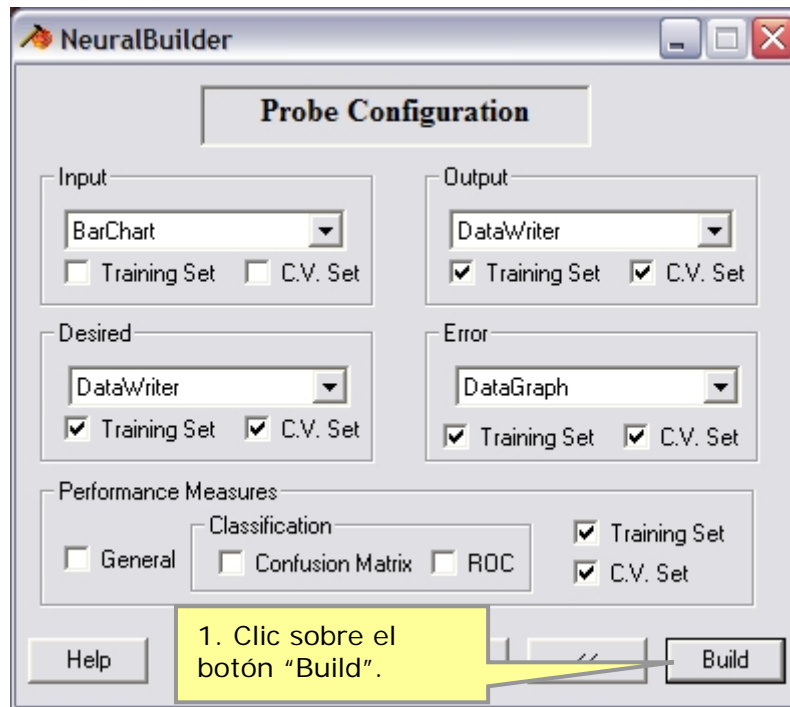


Figura 93. Configurar salida



En la siguiente figura podemos ver un ejemplo de una red creada con el Neuralbuilder:

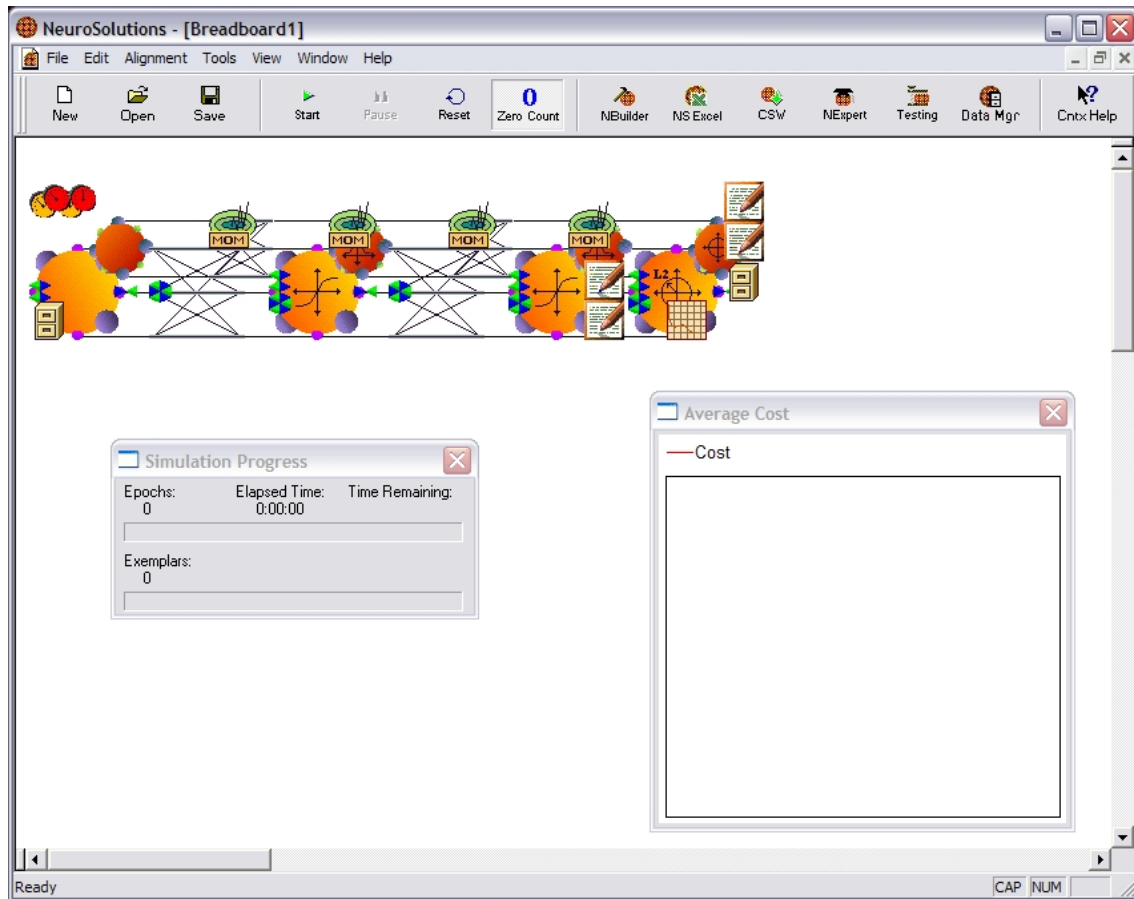


Figura 94. Ejemplo red neurosolutions

A continuación deberemos guardar la red para poder luego utilizarla en la aplicación.

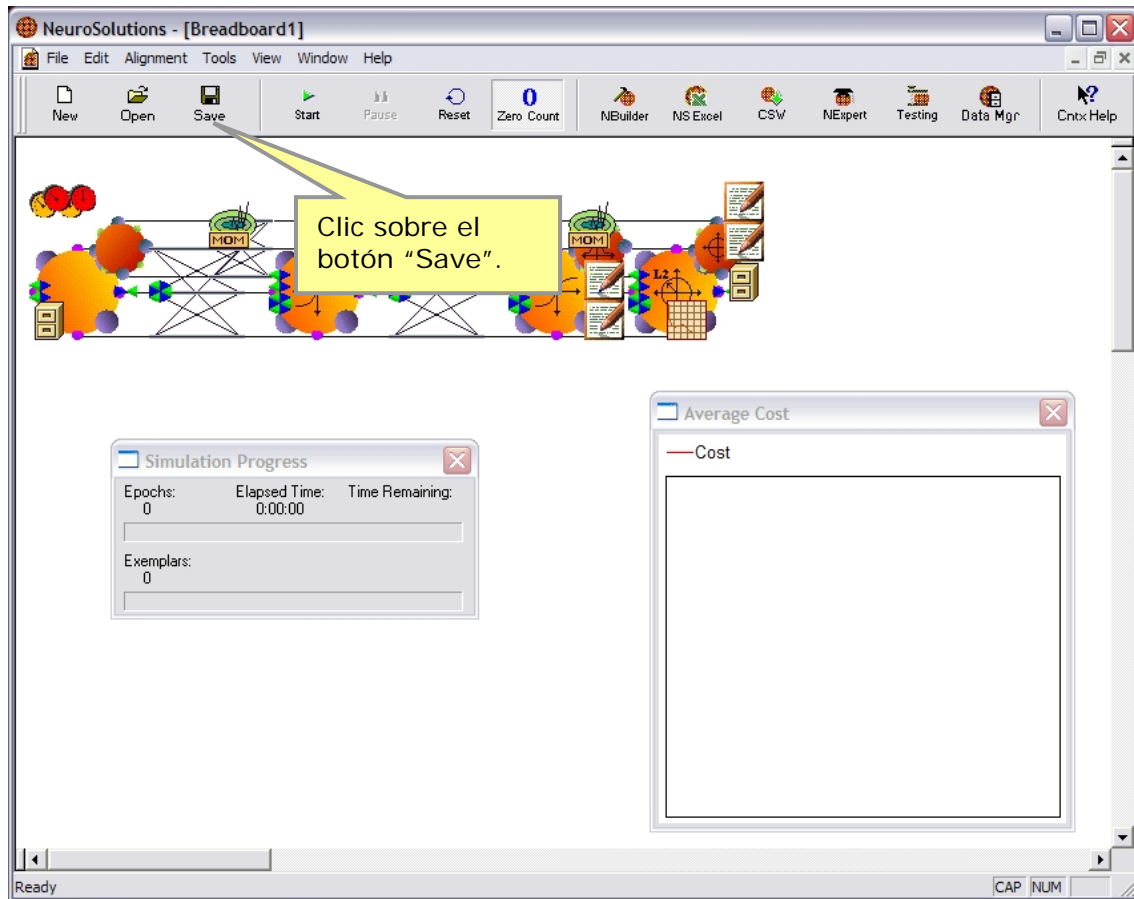


Figura 95. Guardar red de neurosolutions